NAVAL POSTGRADUATE SCHOOL
Monterey, California

THESIS

REMOTE NETWORK ADMINISTRATION OF THE SEANET COMMUNICATION NODE SYSTEM

by

Don C. Murray
and
Christopher L. Pratt

September 1998

Thesis Advisor: Rex Buddenberg
Second Reader: Don Brutzman

Approved for public release; distribution is unlimited.

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. AGENCY USE ONLY *(Leave blank)*	**2. REPORT DATE** September 1998	**3. REPORT TYPE AND DATES COVERED** Master's Thesis

4. TITLE AND SUBTITLE : REMOTE NETWORK ADMINISTRATION OF THE SEANET COMMUNICATION NODE SYSTEM

5. FUNDING NUMBERS

6. AUTHORS
Murray, Don C. and Pratt, Christopher L.

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
Naval Postgraduate School
Monterey, CA 93943-5000

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES)
N/A

10. SPONSORING / MONITORING AGENCY REPORT NUMBER

11. SUPPLEMENTARY NOTES
The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government.

12a. DISTRIBUTION / AVAILABILITY STATEMENT
Approved for public release; distribution is unlimited.

12b. DISTRIBUTION CODE

13. ABSTRACT

Maritime data communications are expensive and of limited capacity. Currently there is no established infrastructure to support Internet connectivity for sea-going vessels. The SeaNet program is investigating maritime networking solutions. One aspect of the SeaNet program is promoting remote network management. Remote network management will provide the maritime research community with a flexible and cost-effective tool for monitoring sea based assets. The objective of this thesis is to investigate remote network management over a satellite connection in support of the SeaNet programs goals.

To research the potential for remote network management, the Naval Postgraduate School has developed its own SeaNet laboratory. This laboratory simulates both the shipboard and shore-based infrastructure of the SeaNet program and conducts remote network management on these components. This thesis discusses the SeaNet program, network management concepts, the NPS SeaNet laboratory, research findings and recommendations for future research. Remote Network Management of the SeaNet Control Node system is possible, however, continued research in this area is needed.

14. SUBJECT TERMS
Network Management, Internet-to-Sea, SeaNet

15. NUMBER OF PAGES
157

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT Unclassified	**18. SECURITY CLASSIFICATION OF THIS PAGE** Unclassified	**19. SECURITY CLASSIFICATION OF ABSTRACT** Unclassified	**20. LIMITATION OF ABSTRACT** UL

REMOTE NETWORK ADMINISTRATION OF THE SEANET COMMUNICATION NODE SYSTEM

Don C. Murray
Lieutenant, United States Navy
B.S., University of Oklahoma, 1991

Christopher L. Pratt
Lieutenant, United States Navy
B.S., Virginia Military Institute, 1990

Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN INFORMATION TECHNOLOGY MANAGEMENT

from the

NAVAL POSTGRADUATE SCHOOL

September 1998

Authors: Don Murray
Don C. Murray

Christopher L. Pratt

Approved by:
Rex Buddenberg, Thesis Advisor

Don Brutzman, Second Reader

Reuben T Harris
Reuben T. Harris
Department of Systems Management

ABSTRACT

Maritime data communications are expensive and of limited capacity. Currently there is no established infrastructure to support Internet connectivity for sea-going vessels. The SeaNet program is investigating maritime networking solutions. One aspect of the SeaNet program is promoting remote network management. Remote network management will provide the maritime research community with a flexible and cost-effective tool for monitoring sea based assets. The objective of this thesis is to investigate remote network management over a satellite connection in support of the SeaNet programs goals.

To research the potential for remote network management, the Naval Postgraduate School has developed its own SeaNet laboratory. This laboratory simulates both the shipboard and shore-based infrastructure of the SeaNet program and conducts remote network management on these components. This thesis discusses the SeaNet program, network management concepts, the NPS SeaNet laboratory, research findings and recommendations for future research. Remote Network Management of the SeaNet Control Node system is possible, however, continued research in this area is needed.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF ACRONYMS

AMSC	American Mobile Satellite Corporation
ARP	Address Resolution Protocol
ARPANET	Advanced Research Projects Agency Network
ASN.1	Abstract Syntax Notation One
AT	Address Translation
bps	bits per second
CLNP	Connectionless Network Protocol
EGP	Exterior Gateway Protocol
FDDI	Fiber Distributed Data Interface
FTP	File Transfer Protocol
HP	Hewlett-Packard
HPOV	Hewlett-Packard OpenView
HSD	High Speed Data
IANA	Internet Assigned Numbers Authority
IBM	International Business Machines
ICMP	Internet Control Message Protocol
IESG	Internet Engineering Steering Group
INMARSAT	International Maritime Satellite
IP	Internet Protocol
IPX	Internetwork Packet Exchange
ISP	Internet Service Provider
JOI	Joint Oceanographic Institute
KVH	Kits van Heyningen
LAN	Local-Area Network
LDEO	Lamont-Dougherty Earth Observatory
MIB	Management Information Base
NAVOCEANO	Naval Oceanographic Office
NCCOSC	Naval Command, Control and Ocean Surveillance Center
NMS	Network Management System
NOPP	National Oceanographic Partnership Program
NPS	Naval Postgraduate School
NRaD	Naval Research and Development
NSF	National Science Foundation
OSI	Open Systems Interconnection
PC	Personal Computer
PDU	Protocol Data Unit
PPP	Point-to-Point Protocol
RARP	Reverse Address Resolution Protocol
RFC	Request For Comment
RMON	Remote Network Management

RTDE	Research, Testing, Development and Evaluation
SAP	SeaNet Advisory Panel
SCN	SeaNet Communications Node
SMI	Structure of Management Information
SNMP	Simple Network Management Protocol
SMTP	Simple Mail Transfer Protocol
SPAWAR	Space and Naval Warfare Systems Command
TCP	Transmission Control Protocol
UDP	User Datagram Protocol
UNOLS	University-National Oceanographic Laboratory Systems
URL	Uniform Resource Locator
WHOI	Woods Hole Oceanographic Institution

I. INTRODUCTION

A. OVERVIEW

This thesis explores the potential for network management of shipboard computer resources from a shore-based site utilizing a satellite link. This project is undertaken in support of the SeaNet program's initiative of extending the Internet to sea-based resources. Specifically, this thesis involves the construction of a laboratory simulating both the shipboard and shore based components of the SeaNet infrastructure. This infrastructure is then monitored via a commercial network management tool. Finally this thesis discusses findings and makes suggestions for continuing research.

B. MOTIVATION

The rapid expansion of the Internet, coupled with advances in network technology, is transforming the maritime research community. While at sea, researchers are less isolated from shore-based computer resources since ship-to-shore network access is possible. Unfortunately, there is no consolidated ship-to-shore networking infrastructure available to support the maritime community. Typical maritime networking efforts are independent, isolated and of limited functionality. Furthermore, maritime communications involve expensive satellite links with relatively low-bandwidth capacities.

Network management is another problematic area for the maritime community. Ideally, network management allows centralized management of computer network resources. Ships typically are limited to two options when addressing network management. The first possibility is for ship and research personnel to manage the network themselves as a collateral duty. While this may save money, most personnel will not possess the requisite knowledge to deal effectively with network problems. Also, this method distracts individuals from their primary duties. The second alternative is that the ship can embark dedicated network management personnel and tools, effectively displacing other researchers. This method is both expensive and inefficient. Most network management tools have the capacity to manage much more than just a shipboard local-area network (LAN).

In order to develop technical solutions for the problems associated with maritime networking, the Office of Naval Research (ONR) has sponsored the SeaNet program. This program is expected to extend the Internet to the sea by establishing the support infrastructure that the maritime community is currently lacking. SeaNet will establish a shore-based operations center, provide satellite communications, develop shipboard communications servers, and support the integration of new technologies [Ref. 1].

C. THESIS OBJECTIVE

In support of SeaNet's objectives, this thesis examines the feasibility of remotely monitoring shipboard computers. Remote network monitoring is expected to relieve the ship of a large number of network administration functions. Remote network monitoring is cost-effective, allowing one individual to monitor many ships from a shore-based site. The ship will no longer need to support an individual simply for the sake of network administration. Bunk and provision space can be allocated to individuals more closely aligned to the ship's primary missions.

D. SCOPE AND METHODOLOGY

This study focuses on the feasibility of conducting network management of a maritime unit from a shore-based station. Specifically the study tests the ability of the shore-based management system to evaluate remote network status. It involves the establishment of a lab consisting of a shipboard node and a shore-based management station connecting over a low-bandwidth satellite connection.

The Naval Postgraduate School (NPS) SeaNet laboratory is supported by multiple partners. Satellite data connectivity and communication equipment is provided by the American Mobile Satellite Corporation (AMSC). The network management station is provided by the Space and Naval Warfare Systems Command (SPAWAR),formerly known as Navy Research and Development (NRaD): Research, Testing, Development, and Evaluation (RTDE). SeaNet Communication Node software is provided by Woods Hole Oceanographic Institution. The simulated shipboard nodes are provided by the NPS.

E. THESIS ORGANIZATION

Chapter II provides an insight into the SeaNet program and its efforts to extend Internet capabilities to sea-based units. Chapter III discusses the general concepts of network management. Specifically, it examines the Simple Network Management Protocol (SNMP). Chapter IV is a description of the NPS SeaNet laboratory. This chapter includes specifics on the components and configurations used in the laboratory. Chapter V reports research findings. It discusses the ability of the NPS network management tool to monitor a remote shipboard station. Chapter VI is the conclusion. It recommends topics for future research in the area of network management.

II. OVERVIEW OF THE SEANET PROGRAM

A. INTRODUCTION

This chapter looks at the SeaNet program. It begins with a brief history of SeaNet. Next, it describes the goals of SeaNet. Third, it lists the organizations that are partners in the SeaNet program and discusses the SeaNet Advisory Panel that guides the decision making process. Finally, it describes the role of NPS as a SeaNet partner.

B. BACKGROUND

> SeaNet is working to create a shore-based and ship-board infrastructure capable of supporting both high and low speed access to the Internet from ships at sea. [Ref. 1].

SeaNet originated in 1995 as a collaborative effort between Woods Hole Oceanographic Institution (WHOI), Lamont-Dougherty Earth Observatory (LDEO), and the Joint Oceanographic Institute (JOI) with funding provided by the National Science Foundation (NSF). This joint effort sought to develop a cost-effective means for sea-going vessels to access the Internet using an INMARSAT B communication system.

This initial effort led to an expansion in the SeaNet group's role. In 1997, the National Oceanographic Partnership Program (NOPP) of the Office of Naval Research (ONR) provided funding for a new SeaNet initiative. The new proposal, "SeaNet: Extending the Internet to the Oceanographic Fleet" involved a $1.478 million dollar investment over two years. This program will allow SeaNet to evaluate both high and low data-rate communication alternatives that will provide Internet connectivity to SeaNet vessels [Ref. 1].

C. GOALS

The SeaNet program looks to extend the Internet-to-sea by establishing a shore-based operations center with Internet Service Provider (ISP) functionality, providing satellite communications to maritime units, developing shipboard communications servers, and supporting the integration of emerging technologies [Ref. 1]. While initially established to support University-National Oceanographic Laboratory System (UNOLS) research vessels, it is desired that this program will ultimately mature into a commercial entity capable of supporting any maritime unit. Once the maritime network service is fully established, it will be relatively simple to add additional nodes (ships) to the network. Just as the Department of Defense's Advanced Research Projects Agency Network (ARPANET) evolved into today's Internet, it is envisioned that eventually the SeaNet program will develop into an ISP for maritime units. Goal capabilities that SeaNet is intended to support include network management, real time data acquisition, remote control applications, client-server database queries, and file transfer.

D. PARTNERS

There are multiple organizations that have joined efforts in the SeaNet program. The following is a list of organizations and individuals from those organizations that represent SeaNet [Ref. 1].

Joint Oceanographic Institutions Dr. Ellen Kappel Liaison/coordination with federal agencies, scientific community, and SeaNet Advisory Panel. http://www.joi-odp.org/
Woods Hole Oceanographic Institution Mr. Andrew Maffei Project coordination; SeaNet Communications Node (SCN) software development. http://www.whoi.edu/
Lamont-Doherty Earth Observatory Mr. Dale Chayes INMARSAT-B procurement; Shipboard systems installation and testing http://www.ldeo.columbia.edu/
Omnet, Inc. Mr. Robert Heinmiller and Ms. Susan Kubany SeaNet operations center; Billing; Value-added services
Naval Postgraduate School Mr. Rex Buddenberg, Shipboard implementation laboratory; Emerging technology planning; NRaD and Navy liaison. http://www.nps.navy.mil/
Naval Command, Control and Ocean Surveillance (NCCOSC): Navy Research and Development Division (NRaD) Technology transfer through NPS
Naval Oceanographic Office (NAVOCEANO) Technical support personnel.

Table 2.1 SeaNet Partners

E. THE SEANET ADVISORY PANEL

To enhance decision making and provide strategic guidance for the SeaNet program, the Joint Oceanographic Institutions organization has developed a SeaNet Advisory Panel (SAP) [Ref. 1]. This panel is made up of member organization representatives and is responsible for providing the following guidance:

- Review and recommend SeaNet unit installations on oceanographic research vessels, and coordinate their usage.
- Recommend priorities for further development for SeaNet.
- Ensure coordination among scientists, ship operators, funding agencies and SeaNet.
- Establish guidelines to evaluate requests for SeaNet equipment and services by individual science projects.

F. THE ROLE OF NPS

The primary role of NPS as a member of the SeaNet program is to perform network research and evaluate emerging technologies for their potential benefits to SeaNet. This particular project evaluates the potential for network management of SeaNet shipboard resources from a land-based site. It is critical that network management is considered an integral part of the SeaNet expansion so that future functionality needs can be considered today.

G. SUMMARY

SeaNet is a program funded by the National Oceanographic Partnership Program (NOPP) of the Office of Naval Research (ONR) to provide computer networking resources and Internet access to vessels at sea. This program has partner organizations that are working together to create an infrastructure for researching these goals. As a partner NPS is conducting research on innovative technologies that will support SeaNet. This thesis addresses remote network management of SeaNet resources.

III. NETWORK MANAGEMENT

A. INTRODUCTION

This chapter will discuss the concept of network management. First, it will present a background for network management and define terms. Second, it will discuss the Simple Network Management Protocol (SNMP) and the Remote Network Management (RMON) extension to SNMP. Third, this chapter will discuss the Management Information Base and, finally, the underlying protocols of SNMP: Internet Protocol (IP) and User Datagram Protocol (UDP).

B. BACKGROUND

Network management is a means of collecting information. Network management software monitors the health and viability of network devices such as hosts, servers, and routers. By monitoring the health of each component, network management tools ensure the well-being of the network as a whole. Ideally, network management is simple, reliable, secure, efficient and inexpensive. The best network management tools are unobtrusive while remaining accessible to the network user. Reporting status changes and detecting network problems is a crucial prerequisite for reliable service.

Network management typically occurs through a series of request/response messages. A Network Management Station (NMS) monitors one or many network devices. The network devices run applications known as agents. The NMS sends requests for information to the agents on a regular cycle. The agents in turn supply the NMS with appropriate responses. An agent may also initiate communications with the NMS when a pre-defined condition is met. These devices and processes are discussed in more detail in the following sections.

C. TERMS

The model for network management includes the following key elements: network management station (NMS), agents, management information base (MIB), and network management protocol.

1. Network Management Station (NMS)

The NMS is a central monitoring host for the network management system. It collects, processes and displays information about the network. The NMS maintains a one-to-many relationship between itself and a set of managed network devices. The NMS is able to get and set variables in the managed devices, and can also receive messages from the agents. It is possible to have several NMS hosts, each of which can manage all (or a subset) of the stations in the configuration.

An NMS has the following components:

- a set of management applications for data analysis and fault recovery.
- an interface by which the network NMS may monitor and manage the network.
- the capability of actively conducting monitoring and control of remote elements in the network.
- a database of information extracted from the managed entities in the network.

The specific NMS software used in this thesis project is the HP OpenView program running on a Hewlett-Packard Apollo Series 700 computer. More information on this system is provided in Chapter IV.

2. Agent

Agent software runs on network devices, allowing them to be managed by the NMS. Agents respond to `get` and `set` messages sent by the NMS and notify the NMS if any predefined conditions are met. Network devices which include running agents may be physical (such as a host, router, or bridge) or logical (such as a network service or application program).

Although a network typically contains many agents, this thesis project looks only at managing one network device, a personal computer (PC). The PC runs a Simple Network Management Protocol (SNMP) agent under the Red Hat Linux operating system, version 5.0.

3. Management Information Base (MIB)

The MIB is a database of network and network device information. Network devices are managed by representing each device as a set of objects. Each object is essentially a data variable that represents one aspect of the managed device. The collection of these objects is simply a database that is referred to as a MIB. Maintaining the MIB is the responsibility of the NMS. MIB objects are divided into classes, allowing certain classes to be applied to different network devices. MIB-II, the current MIB standard, is discussed in depth in section E of this chapter.

4. Network Management Protocol

The network management station and network agents are linked by a network management protocol. This protocol defines the structure of information transferred over the network. The protocol most often used for the management of Transmission Control Protocol/Internet Protocol (TCP/IP) networks is the Simple Network Management Protocol (SNMP).

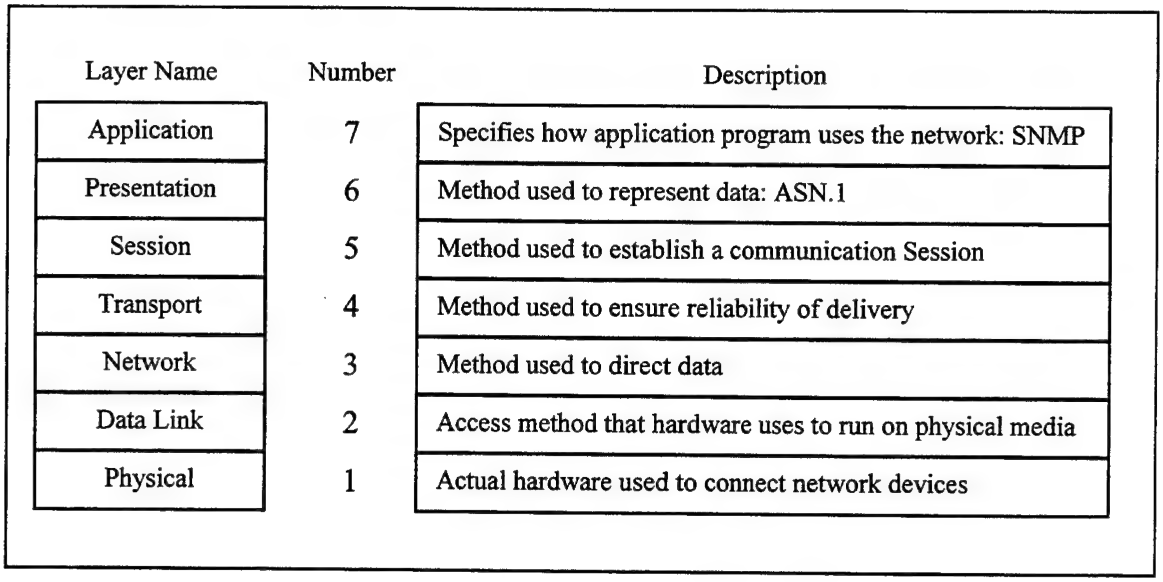
D. SIMPLE NETWORK MANAGEMENT PROTOCOL

This section discusses the Simple Network Management Protocol (SNMP) versions 1, 2 and 3.

1. Overview

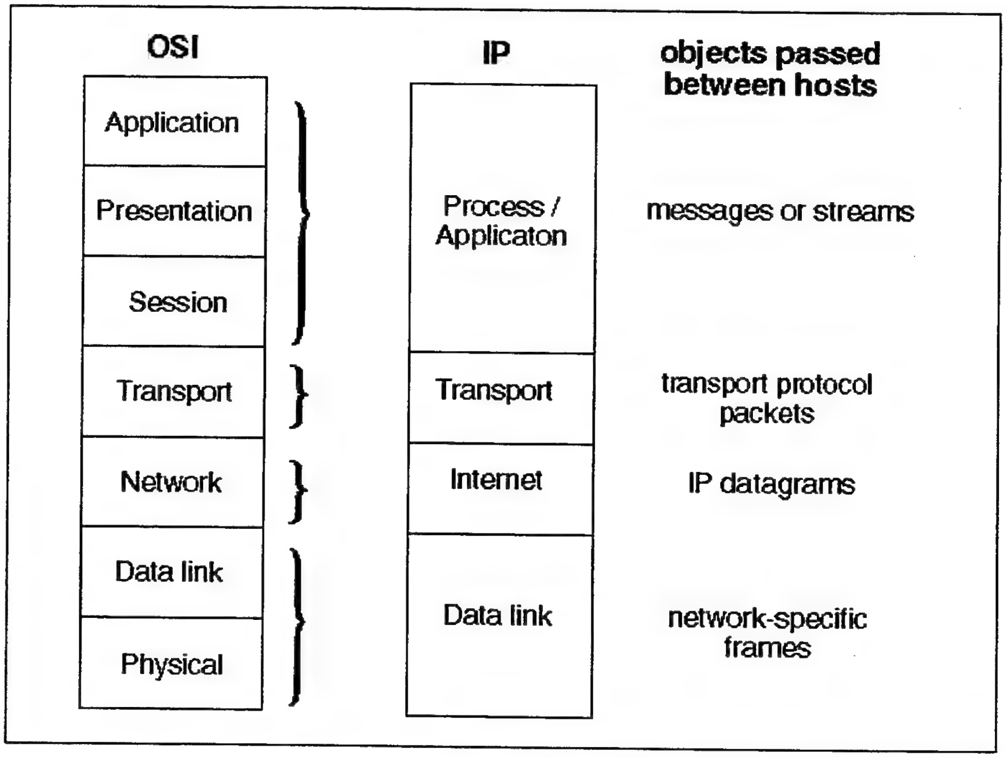
The leading solution for network management is the Simple Network Management Protocol (SNMP). "The SNMP protocol defines exactly how a manager communicates with an agent" [Ref. 2]. Outlined in Request for Comment (RFC) 1157, SNMP version I was adapted as the standard for TCP/IP-based Internets in 1989. SNMP is a simple, inexpensive and mature network management protocol with an extensive support base.

SNMP operates at the application layers of both the Open Systems Interconnection (OSI) and Internet Protocol (IP) models for networking. Figure 3.1 depicts the OSI networking model while Figure 3.2 depicts the correspondence between the OSI model and the widely implemented IP model. Like all application layer software, SNMP transports data over the network using conventional transport protocols. Unfortunately, data loss failures due to a variety of causes can prevent packets from traveling to or from a host application layer. This means that network management may be lost when it is most needed, (i.e. when some part of the network fails). 'In practice, however, using an application level protocol for network management works well for two reasons. First, in the case of hardware failure, the level of the protocol does not matter – the NMS can communicate with those devices that remain operating and use success (or failure) to help locate the problem. Second, using conventional transport protocols means the NMS's packets will be subject to the same conditions as normal traffic. Thus, if delays are high, a NMS will find out immediately [Ref. 2].



Layer Name	Number	Description
Application	7	Specifies how application program uses the network: SNMP
Presentation	6	Method used to represent data: ASN.1
Session	5	Method used to establish a communication Session
Transport	4	Method used to ensure reliability of delivery
Network	3	Method used to direct data
Data Link	2	Access method that hardware uses to run on physical media
Physical	1	Actual hardware used to connect network devices

Figure 3.1 The OSI Model for Networking



OSI	IP	objects passed between hosts
Application Presentation Session	Process / Applicaton	messages or streams
Transport	Transport	transport protocol packets
Network	Internet	IP datagrams
Data link Physical	Data link	network-specific frames

Figure 3.2 Correspondence between OSI and IP protocol layer models, and objects passed between corresponding host layers. [Ref. 3]

2. SNMP Protocol Data Unit

The SNMP message, known as a Protocol Data Unit (PDU), is used to exchange information between the NMS and the agents. There are five types of PDUs: `GetRequest`, `GetNextRequest`, `SetRequest`, `GetResponse` and `Trap`. The structure of the generic PDU is shown in Figure 3.3.

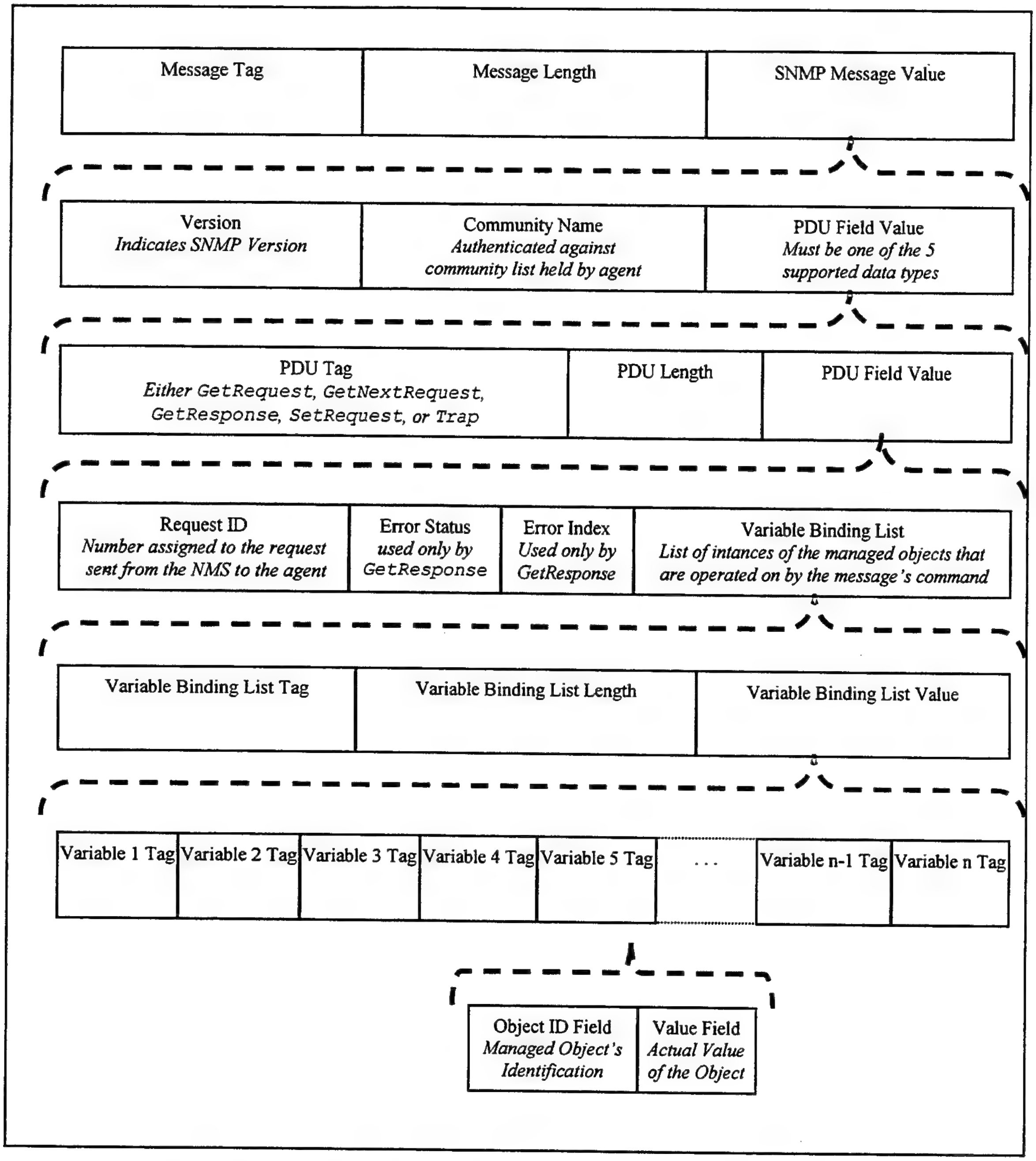

Message Tag
Message Length
SNMP Message Value
Version
Indicates SNMP Version
Community Name
Authenticated against community list held by agent
PDU Field Value
Must be one of the 5 supported data types
PDU Tag
Either GetRequest, GetNextRequest, GetResponse, SetRequest, or Trap
PDU Length
PDU Field Value
Request ID
Number assigned to the request sent from the NMS to the agent
Error Status
used only by GetResponse
Error Index
Used only by GetResponse
Variable Binding List
List of intances of the managed objects that are operated on by the message's command
Variable Binding List Tag
Variable Binding List Length
Variable Binding List Value
Variable 1 Tag
Variable 2 Tag
Variable 3 Tag
Variable 4 Tag
Variable 5 Tag
. . .
Variable n-1 Tag
Variable n Tag
Object ID Field
Managed Object's Identification
Value Field
Actual Value of the Object

Figure 3.3 The generic Simple Network Management Protocol (SNMP) Data Unit (PDU)

a. *The GetRequest PDU*

The NMS issues the GetRequest PDU to get MIB values from an agent. The receiving SNMP agent's response to a GetRequest PDU is a GetResponse PDU containing the same request-id and the pertinent data. The GetRequest operation is atomic, meaning that either all of the values are retrieved or none are. If the responding agent is able to provide values for all of the variables listed in the incoming variable bindings list, then the GetResponse PDU includes the variable-bindings field, with a value supplied for each variable. If any one the variable values cannot be supplied, then no values are returned and an error code is generated.

The request and response behavior of SNMP PDUs is shown in Figure 3.4.

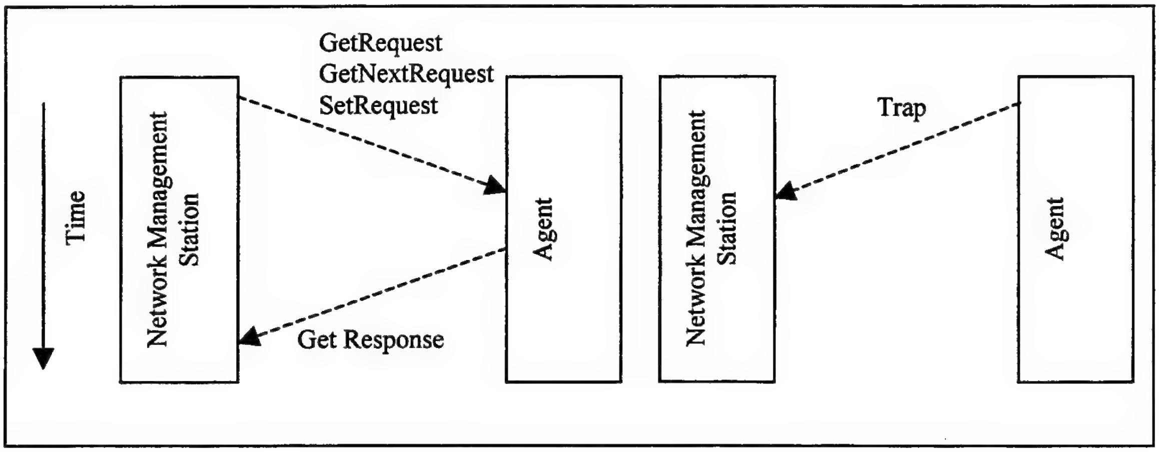


Figure 3.4 SNMP PDU request and response behavior, requested and trapped.

b. *The GetNextRequest PDU*

Almost identical in structure to the GetRequest PDU is the GetNextRequest PDU. In the GetNextRequest PDU, for each variable, the agent returns the value of the object instance that is next in lexicographical order (not just the next object). This allows the NMS to retrieve information from an agent without knowing exactly which MIB objects the agent supports. It also provides an efficient mechanism for searching a table whose entries are unknown. Note that `GetNextRequest` only looks for the next object instance that occurs lexicographically after the identifier supplied. There is no requirement that the supplied identifier represent an actual object. Like `GetRequest`, `GetNextRequest` is atomic. Also like `GetRequest`, improper requests will generate `GetResponse` error codes.

c. *The* ***`SetRequest`*** *PDU*

The `SetRequest` PDU is used by the NMS to change the value of a MIB variable at the agent. The agent responds to the `SetRequest` PDU with a `GetResponse` PDU that reflects the updated information. As before, the `SetRequest` operation is atomic, meaning that either all of the variables are updated or none is. Bad requests will result in `GetResponse` error messages from the agent.

A major flaw of SNMP is that it provides no means of security. "Specifically, there is no capability to authenticate the source of a management message nor any capability to prevent eavesdropping" [Ref. 4]. Although an SNMP message must contain a valid community name, this name may be learned by anyone who intercepts an SNMP message. SNMP is extremely vulnerable to attacks that modify network device configurations using the `SetRequest` PDU. For this reason, many vendors choose not to employ the `SetRequest` PDU in their implementations.

d. *The GetResponse PDU*

The GetResponse PDU is used only as a response to GetRequest and GetNextRequest queries. If the agent is able to find values for all instances requested in the Get PDU it will generate a GetResponse PDU with the appropriate values. The GetResponse PDU looks exactly the same as the originating Get PDU except that the PDU tag indicates a GetResponse PDU and the Value Field will have actual values.

If the agent is unable to process the Get PDU for any reason, it will return an error code. Since the GetResponse PDU is atomic it cannot process some values while providing error codes for others. Any single error will result in an overall error code. The following error conditions are possible:

- noSuchName: An object named in the variable bindings field does not match any object identifier in the relevant MIB view.

- tooBig: responding size of the resulting GetResponse PDU exceeds local limitations.

- genErr: Responding entity is not able to supply a value for at least one of the object for some reason.

- badValue: badValue is used only in response to the SetRequest PDU. badValue is returned if the SetRequest PDU contains at least one pairing of variable name and value that is inconsistent. The inconsistency might be in the type, length, or actual value of the supplied value

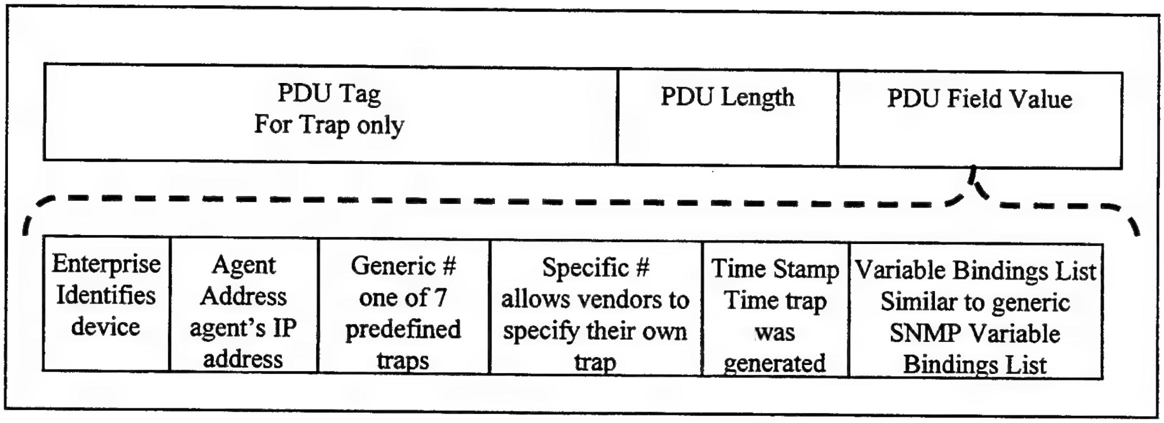
e. *The Trap PDU*

The Trap PDU is used by the agent to send unsolicited MIB values to the NMS. Traps occur when MIB values that are being monitored by the agent exceed a preset threshold. Each station is responsible for notifying the management station of any unusual event; for example, the agent crashes and is rebooted, a link fails, or an overload condition as defined by the packet load crosses some threshold.

Once a management station is alerted to an exception condition, it may choose to take some action. At this point, the management station may direct polls to the agent reporting the event and perhaps to some nearby agents in order to diagnose any problem and to gain more specific information about the exception condition.

Trap directed polling can result in substantial savings of network capacity and agent processing time. In essence, the network is not made to carry management information that the management station does not need, and agents are not made to respond to frequent requests for uninteresting information.

The PDU Field Value for a trap differs from that of the generic PDU. The trap structure is shown in Figure 3.5



PDU Tag For Trap only	PDU Length	PDU Field Value

Enterprise Identifies device	Agent Address agent's IP address	Generic # one of 7 predefined traps	Specific # allows vendors to specify their own trap	Time Stamp Time trap was generated	Variable Bindings List Similar to generic SNMP Variable Bindings List

Figure 3.5 The `Trap` PDU

3. SNMP Version 2

SNMPv2 was adapted as an upgrade to SNMP in 1993 (later revised in 1996) [Ref. 4]. SNMPv2 enhances SNMP in three specific areas: the structure of management information, manager-to-manager communications, and protocol operations. SNMPv2 adds several new data types and enhances documentation associated with MIB objects. The SNMPv2 MIB contains information about the operation of SNMPv2 itself and information about the configuration of a SNMPv2 NMS or agent. While the 1993 version of SNMPv2 included security enhancements, these enhancements were problematic and dropped altogether in the 1996 revision.

SNMPv2 supports several additional transport protocols such as AppleTalk, Internetwork Packet Exchange (IPX), and the Open Systems Interconnection (OSI) Connectionless Network Protocol (CLNP) stacks. SNMPv2 also introduces two new PDUs: `GetBulkRequest` and `InformRequest.` `GetBulkRequest` allows for the collection of large amounts of data without retrieving each data field one at a time as the `GetNextRequest` PDU does [Ref. 5]. `InformRequest` facilitates NMS-to-NMS communications. The manager-to-manager reporting capabilities allow multiple NMS hosts to be configured in a hierarchical fashion, with several lower-tier NMS hosts reporting to a single higher-tier NMS.

Additionally, SNMPv2 implements mechanisms to assure that multiple NMS communications are conducted properly. The SNMPv2 Structure of Management Information (SMI) was modified to accommodate 64-bit counters (SNMPv1 uses 32-bit counters). Error reporting codes are expanded and agents are allowed to process partial requests for information.

Despite these improvements, the public reception of SNMPv2 has been less than positive. The initial security corrections (later dropped) were perceived as complicated and ineffectual. Additionally, agent configuration was difficult and the administration of SNMPv2 was a great deal more complex than SNMPv1. In short, SNMP was no longer simple. Currently, SNMPv2 is rarely used.

4. SNMP Version 3

Current initiatives are underway to introduce a third version of SNMP. SNMPv3 will combine the functionality of SNMPv1 and experimental derivations of SNMPv2 (SNMPv2p, SNMPv2u and SNMPv2*) [Ref. 6]. SNMPv3 is expected to be simultaneously secure, extendable, and easy to administer and configure. SNMPv3 includes 3 modules:

- The Message Processing and Control module handles SNMP message creation and parsing functions, and also determines if proxy handling is required for any SNMP message [Ref. 7].
- The Local Processing module performs access control for variable binding data, processing that data and trap processing [Ref. 7].
- The Security module provides authentication, and encryption functions, and checks the timeliness of certain SNMP messages [Ref. 7].

Currently, SNMPv3 is in Internet draft form. The SNMPv3 working group is expected to complete all core specifications and forward documents to the Internet Engineering Steering Group (IESG) for consideration as Draft Standard RFCs in October 1998 [Ref. 8]. Consequently, all further mention of SNMP in this thesis refers to SNMP version 1.

E. THE MANAGEMENT INFORMATION BASE (MIB)

1. Background

The MIB is the collection of all the objects that the NMS can manage. MIB objects are the unique MIB names and MIB instances are the values for those names. For example, in the key=value pair x=4, x is the object and 4 is the instance. The NMS collects specific instances for each object and device, and maintains those instances in a database known as the MIB database. The MIB database is constantly updated using SNMP. The NMS issues MIB objects to the agents using the `Get` commands. The agents respond to the `Get` Commands with `GetResponse` messages containing the requested MIB instance or through issuing Trap messages (which contain MIB instances). Either way the MIB database is populated with instance values.

The MIB is defined separately from SNMP. In other words, SNMP provides a framework for transferring MIB objects but does not define what the objects should be. This flexible design allows users to define and implement their own MIB objects as required. The MIB structure can be thought of as a hierarchical organization chart (see Figure 3.6). MIB objects are assigned specific names and places within the chart to ensure uniqueness and prevent redundancy. The MIB hierarchy is divided into classes.

These classes are structured according to who is creating the MIBs and what devices the MIBs are designed to manage.

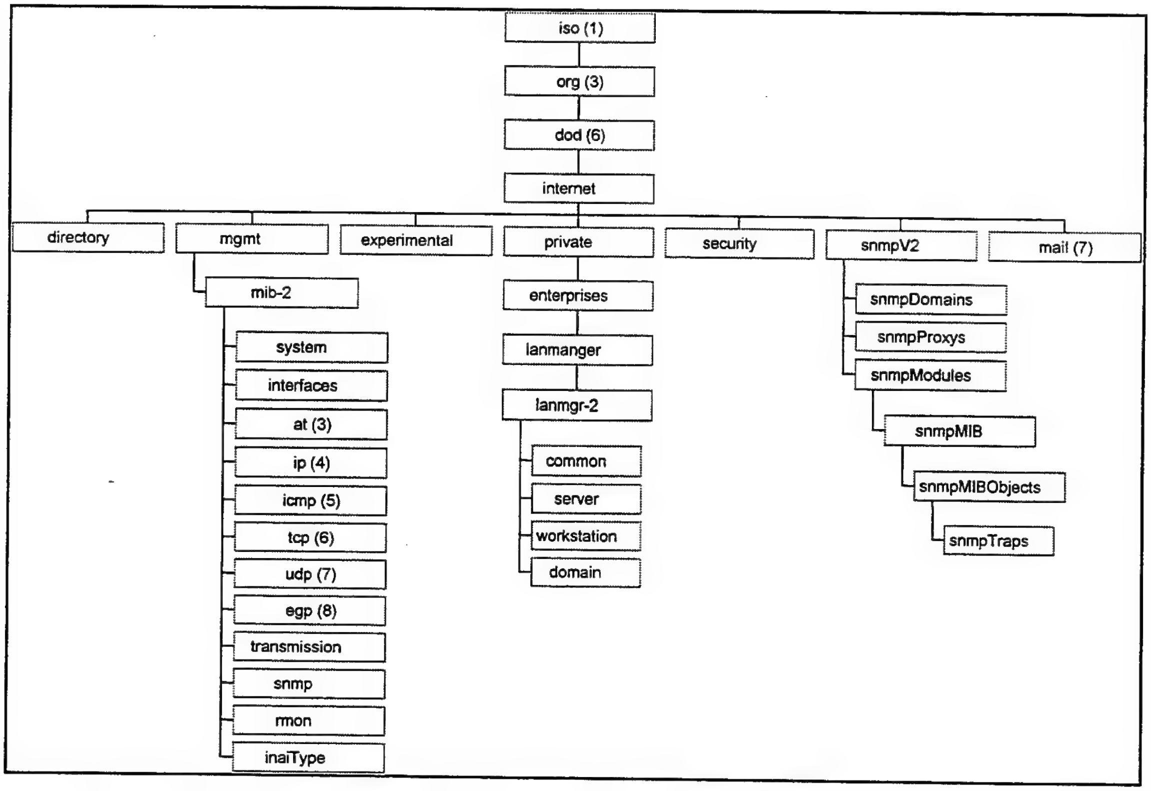


Figure 3.6 The Management Information Base (MIB) Hierarchical Structure

Objects in a MIB are defined with the Abstract Syntax Notation One (ASN.1) naming scheme, which assigns each object a long prefix that guarantees the name will be unique. For example, an integer that counts the number of IP datagrams a device has received is named: iso.org.dod.internet.mgmt.mib.ip.ipInReceives [Ref 2].

Furthermore, when the object name is represented in an SNMP message, each part of the name is assigned an integer. Thus, in an SNMP message, the name of ipInReceives is: 1.3.6.1.2.1.4 [Ref. 2].

It must be noted that MIB "objects" are simple variables and not really objects at all. A MIB object may be a simple two-dimensional variable (indicating only MIB object name and network device name), or an array or table (such as an IP routing table).

This chapter will now discuss two widely accepted sets of MIB objects: MIB-II and RMON.

2. MIB-II

MIB-II is generally accepted as the standard or core MIB. It contains information deemed essential for either fault or configuration management [Ref. 4]. Originally defined in RFC 1213, MIB-II expands on a 1988 MIB known as MIB-I. Originally released in 1990, MIB-II was later revised in 1991. MIB-II contains 10 groups and 171 objects.

The MIB-II object groups are as follows: System, Interfaces, Address Translation (AT), Internet Protocol (IP), Internet Control Message Protocol (ICMP), Transmission Control Protocol (TCP), User Datagram Protocol (UDP), Exterior Gateway Protocol (EGP), Transmission, Simple Network Management Protocol (SNMP). Each of these object groups are now examined in further detail.

a. The Systems Group

The Systems group contains descriptive information about a managed device such as point or contact information (sysContact) and the host's operating system (sysDescr). The Systems Group contains seven objects.

b. The Interfaces Group

The Interfaces group contains generic information about the managed device's network interfaces. This information includes interface product name and version (ifDescr), the speed of the interfaces' current data rate capacity (ifSpeed), the number of octets transmitted (ifOutOctets) and the number of inbound packets that were discarded due to errors (ifInErrors). The Interfaces Group contains twenty-three objects.

c. The Address Translation (AT) Group

The Address Translation (AT) group provides information on how the device maps a physical address from a network (IP) address. When MIB-II was written, it was decided that the MIB-I implementation of the AT group was too limiting and that AT information was included in other places in MIB-II. Consequently, the AT group was given a depreciated status. The AT group is included in MIB-II, however, to provide compatibility with existing MIB-I implementations. The AT group contains three objects.

d. The Internet Protocol (IP) Group

The Internet Protocol (IP) group provides information on IP operations. These operations include the total number of input datagrams successfully delivered to IP user protocols (ipInDelivers), the Subnet mask associated with the IP address of the managed device (ipAdEntNetMask), and the IP address of the next hop for a given route (ipRouteNextHop). The IP group contains forty-eight objects.

e. The Internet Control Message Protocol (ICMP) Group

The Internet Control Message Protocol (ICMP) group contains statistics on ICMP input and output operations. These statistics include the total number of ICMP messages received (icmpInMsgs), the Number of ICMP Echo Reply messages received (icmpInEchoReps), and the number of ICMP Address Mask Reply messages sent (icmpOutAddrMaskReps). The IP group contains twenty-six objects.

f. The Transmission Control Protocol (TCP) Group

The Transmission Control Protocol (TCP) group contains statistics about TCP operations. These statistics include the device's TCP retransmission algorithm (tcpRtoAlgorithm), the total number of retransmitted segments (tcpRetranSegs), and the total number of TCP segments received in error (tcpInErrors). The TCP group contains twenty-one objects.

g. The User Datagram Protocol (UDP) Group

The User Datagram Protocol (UDP) group contains information relevant to the UDP operations. This information includes the total number of UDP packets delivered (udpInDatagrams) and the total number of UDP datagrams received which had no application at the desired port (udpNoPorts). The UDP group contains eight objects.

h. The Exterior Gateway Protocol (EGP) Group

The Exterior Gateway Protocol (EGP) group contains information concerning the operation of EGP. The EGP group includes eight objects which comprise the EGP Neighbor Table, providing information about each of the neighbor gateways known to the device. The EGP group contains twenty objects.

i. The Transmission Group

The Transmission group, also known as the dot3 group, contains information about the transmission schemes and access protocols at each system interface [Ref. 4]. Based on the transmission media underlying each interface on a system, the corresponding portion of the Transmission group is mandatory for that system. When Internet-standard definitions for managing transmission media are defined, the Transmission group is used to provide a prefix for the names of those objects [Ref. 9]. The sixteen Transmission MIBs are currently defined. These groups include X.25 packet technology, Token Ring, Frame Relay, and Fiber Distributed Data Interface (FDDI) technology among others.

j. The SNMP Group

The SNMP group contains information about SNMP operations. This information includes the total number of messages delivered to the agent from the transport service (snmpInPkts) and the total number of SNMP `SetRequest` PDU's accepted and processed by the agent (snmpInSetRequests). There are twenty-eight objects in the SNMP group.

3. Remote Network Management

First issued in 1991, Remote Network Management (RMON) acts as a supplement to MIB-II. RMON extends SNMP's capabilities to include the management of local-area networks (LANs). RMON agents collect statistics about the network itself, not just the network devices. "Without RMON an NMS has difficulty constructing a profile of the network activity" [Ref. 10].

Since RMON monitors the network rather than the device, only one RMON agent (called an RMON probe) is required for the network. Thus, a LAN consisting of many SNMP agents may contain only one RMON probe.

RFC 1757, Remote Network Monitoring Management Information Base, lists five primary goals that can be realized with RMON implementation [Ref 10]:

- Off-line operation: allowing the RMON agent to collect data even if the connection between it and the NMS is interrupted (This RMON implementation goal will be of particular interest to future research projects). Off-line operation allows the RMON probe to continue to collect information on the local-area network (LAN) even while it is disconnected from the NMS. This collected information is then passed to the NMS when connection occurs.

- Preemptive monitoring: allowing the agent to preempt its normal monitoring tasks to notify the NMS of a failure.

- Problem detection and reporting: agent can detect and report specific error conditions.

- Value-added data: Agent can collect and retrieve management data for the NMS.

- Multiple managers: agent must support concurrent communications with multiple NMS.

In 1996, RMON's capabilities were extended with a version known as RMON-II. RMON-II does not replace RMON-I but rather complements it. Together the two RMON versions can monitor each of the network layers. While RMON-I concentrates on monitoring the bottom (physical and network) layers of the IP stack, RMON-II focuses on the transport and application layers.

F. THE UNDERLYING TRANSPORT PROTOCOLS

This section will examine the structure of the Simple Network Management Protocol (SNMP) and its underlying transport protocols, the User Datagram Protocol (UDP) and the Internet Protocol (IP) (See Figure 3.7). This examination will help to explain the behavior of SNMP.

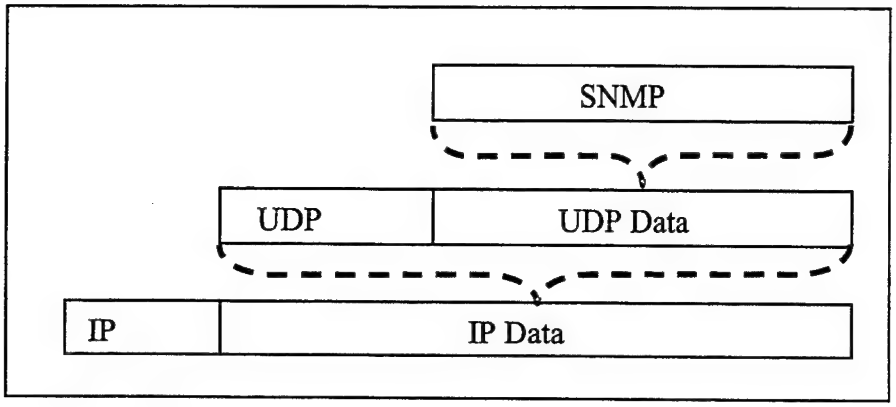


Figure 3.7 SNMP's Underlying Protocols [Ref 2]

These three protocols (SNMP, UDP, and IP) correspond to the Application, Transport and Network layers of the IP network model (see Figure 3.8).

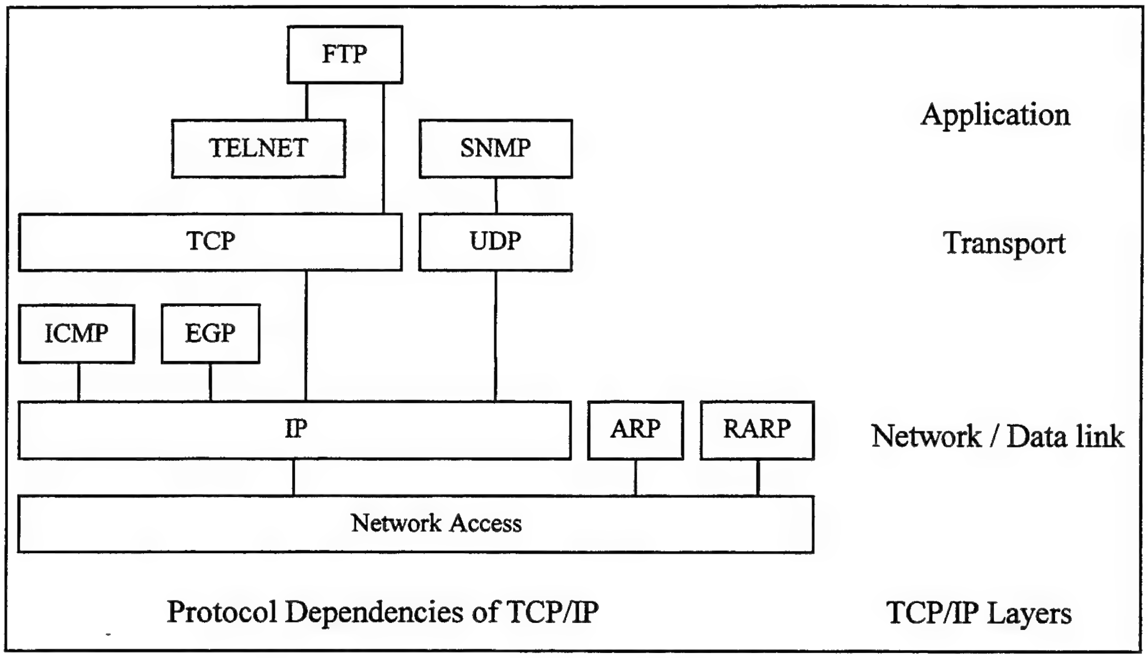


Figure 3.8 The relationship of SNMP to TCP/IP layers [Refs. 11 and 12]

1. Internet Protocol (IP)

The Internet Protocol (IP) acts as the network layer for SNMP. IP is a connectionless and unreliable (i.e. best-effort) protocol. The IP datagram header contains the data to route and deliver a packet to its destination address on the internet.

The IP datagram header consists of fourteen fields of various lengths. The maximum size of an IP datagram (including header) is 1500 bytes. Larger datagrams are broken into fragments and transmitted separately across the internet. The IP Fragment Offset field provides a means of ensuring the fragmented packets are reassembled in proper order. Details on IP version 4 can be found in RFC 791. The IP datagram format is shown in Figure 3.9.

<table>
<tr><td>IP Version
4 bits
Identifies IP version</td><td>IP Header Length
4 bits
In 32 bit words
Min value is 5</td><td>Type of Service
8 bits
Specifies reliability, precedence, and throughput</td><td colspan="2">IP Total Length
16 bits
Total length of datagram including header
Measured in octets</td></tr>
<tr><td colspan="3">IP Identification
16 bits
Datagram originator ID
Uniquely identifies each datagram</td><td>Fragment Flags
3 bits</td><td>Fragment Offset
13 bits
Indicates fragment's position in the datagram</td></tr>
<tr><td colspan="2">Time to Live
8 bits
Counter measured in router hop increments</td><td>Protocol
8 bits
Transport protocol number</td><td colspan="2">Header Checksum
16 bits
Calculated when IP is sent and then recalculated when received
If not equal then the datagram is discarded</td></tr>
<tr><td colspan="5">Source Address
32 bits
Senders IP address</td></tr>
<tr><td colspan="5">Destination Address
32 bits
Destination IP address</td></tr>
<tr><td colspan="3">IP Options
Variable length
Optional
used primarily for network testing or debugging</td><td colspan="2">Padding
Variable length
Zeros only
added to make IP header end on a 32-byte boundary</td></tr>
<tr><td colspan="5">IP Data</td></tr>
</table>

Figure 3.9 The Internet Protocol (IP) datagram.

2. User Datagram Protocol (UDP)

User Datagram Protocol (UDP) typically acts as the transport protocol underlying SNMP. UDP is used to deliver packets to a specific port address at given host computer. In contrast to the more widely used Transport Control Protocol (TCP), UDP is simple and unreliable (i.e. best effort). UDP is unreliable in that it does not transmit a "receipt" packet to the transmitting address; therefore a sender never knows if the intended recipient actually received the packet or not. In addition, UDP does not provide any data flow control mechanism nor does it deal with packet loss or corruption. "Thus, UDP messages can be lost, duplicated, or arrive out of order" [Ref. 13]. Despite these drawbacks, UDP is often utilized, primarily for its simplicity. Since SNMP does not require all the services of TCP, the simplified packet header of UDP eliminates unnecessary overhead.

Because SNMP is typically used to administer reliable, low-delay LANs, users may not encounter the shortcomings of UDP. "Application programs that rely on UDP work well in local environment but fail in dramatic ways when used in a larger TCP/IP Internet." [Ref. 13].

The UDP header consists of four 16-bit fields. These fields identify the source and destination ports of each packet, the message length, and provide a checksum. The UDP Source Port and UDP Checksum fields are optional. These fields are shown in Figure 3.10.

UDP Source 16 Optiona Specifies the port to which replies should	UDP Destination 16 Indicate the port receiving process is Port 161 reserved for SNMP net Port 162 reserved for SNMP
UDP Message 16 Contains datagram octets including Minimum value is	UDP 16 Optiona
UDP	

Figure 3.10 The UDP Datagram

Port numbers correspond to specific communications processes within a host node. Peer processes are able to communicate with each other over a data connection by addressing their packets to a specific port and address combination. This port is monitored at the peer host process. IP addressed port 161 is reserved for SNMP network monitor messages and port 162 is reserved for SNMP traps. Some port numbers are reserved y the Internet Assigned Numbers Authority (IANA), ensuring port addresses are uniquely assigned [Ref. 14]. An example of UDP demultiplexing to appropriate ports in an addressed host is shown in Figure 3.11.

Another consideration for UDP is the way it is handled by firewalls. Typically, firewalls are configured to not pass UDP packets. In order to monitor devices through firewalls using SNMP, the firewall needs to be configured to allow UPD packets that correspond to a specific UDP port addresses such as 161 and 162. Of course, firewalls need not be configured to accept all SNMP packets, but only those that the NMS needs. One possible vulnerability from opening SNMP UPD ports through a firewall is that SNMP packets can be used for a denial-of-service type attack.

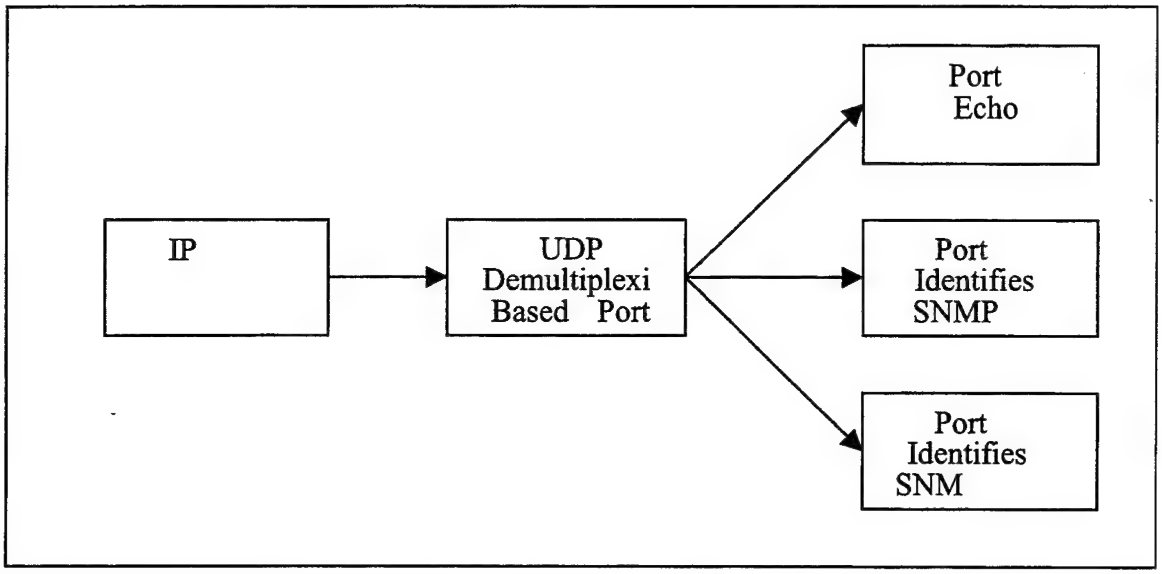


Figure 3.11 UDP Demultiplexing Operations

G. SUMMARY

Network management is a means of monitoring the condition of the network. Network management requires a Network Management Station, agents, a management information base (MIB) and a network management protocol. Essential MIB objects are defined in MIB-II and its supplement, RMON. Simple Network Management Protocol (SNMP) version 1 is the most widely used network management protocol. SNMP is supported by the underlying protocols of UDP and IP.

IV. THE NPS SEANET LABORATORY

A. INTRODUCTION

This chapter describes the NPS SeaNet laboratory. It introduces the equipment of a typical SeaNet vessel that the NPS SeaNet laboratory seeks to emulate, provides configuration information, and describes operation of the laboratory equipment.

B. SEANET VESSEL RESOURCES

Regarding computer and communication resources, there is no common configuration for maritime vessels. Exact configurations vary from ship to ship and from mission to mission. One common component for SeaNet vessels does exist, however, the SeaNet Control Node System. This system is the interface that bridges the vessel's LAN to shore based resources. This section discusses the resources of a single research vessel (the Atlantis) and provides a description of the SeaNet Communication Node (SCN) software.

1. Computer and Satellite Communication Resources

Table 4.1 provides an overview of the computer and satellite resources typically available on the WHOI vessel Atlantis:

Computer Resources
Main Hub, Central Fibronics Gigahub capable of 100 Mb/s
Servers, Intel Pentium based IBM PC's running Red Hat Linux version 4.1
10 Base T Ethernet cabling throughout ship
Nine separate collision domains
TCP/IP protocol suite use
Class C network IP address with domain name "atlantis.whoi.edu"
50 IP nodes available
Support for multiple Operating Systems i.e., WinNT, MacOS, Solaris, Unix, Linux (Red Hat distribution)
Satellite Communication Resources
INMARSAT A, Magnavox MX 2400, 9.6K baud data transfer rate
INMARSAT B High Speed Data (HSD), Nera Saturn, 64K baud data transfer rate

Table 4.1 Vessel Atlantis Computer and Satellite Communication Resources

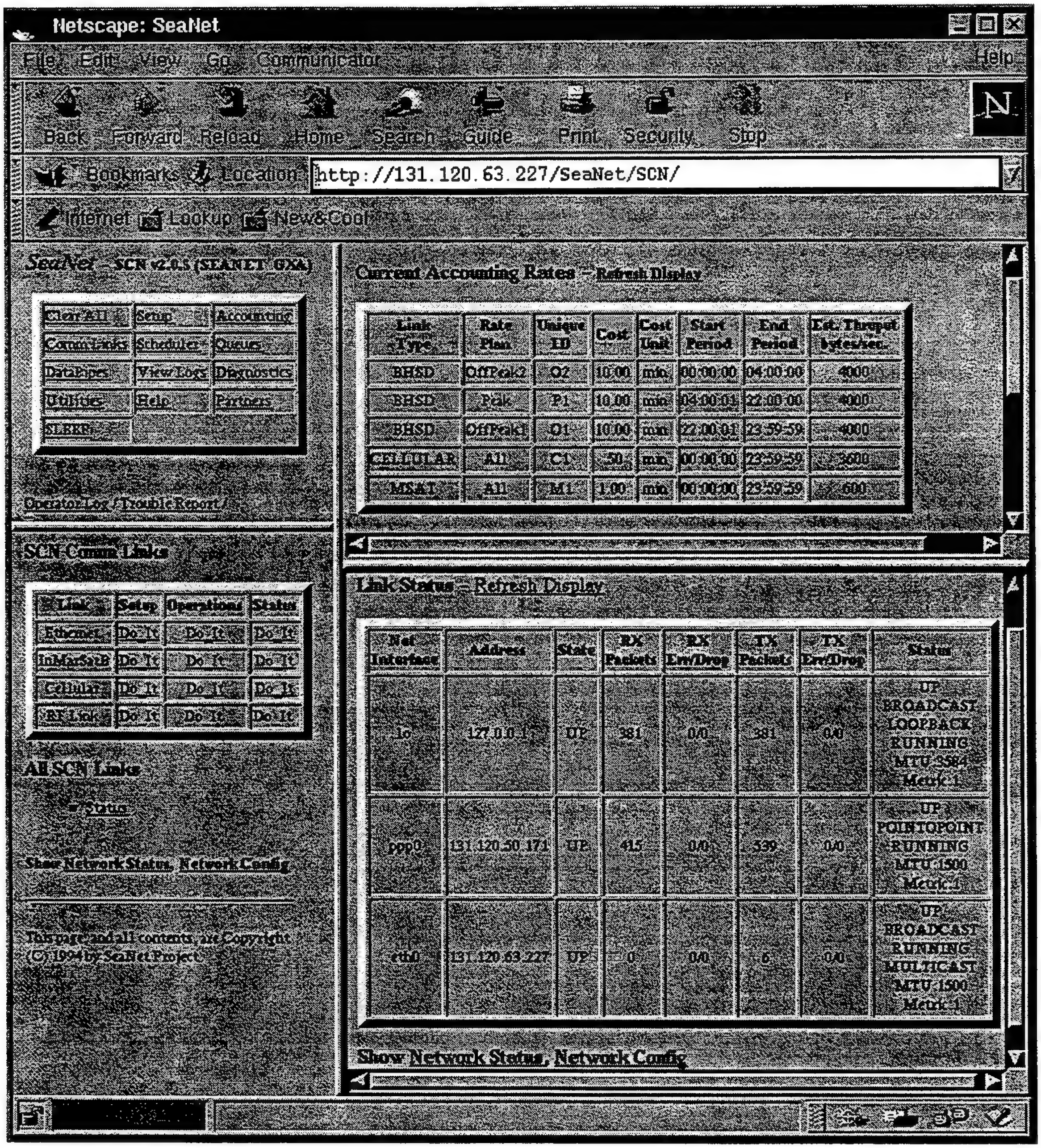
2. Software Resources

The SeaNet Communications Node software is a browser based program developed by WHOI and designed to run on the shipboard SCN system. The SCN software performs the following functions:

- Control and configuration of communication links (i.e. INMARSAT, AMSC, Cellular).
- Satellite connection account based cost information.
- Batch file transfer capability.
- Web mirroring.
- Software programming resources for enhanced future functionality.

Figure 4.1 shows Netscape Navigator for Linux with the SeaNet Communications Node software running. This section will briefly discuss two specific functions of the SCN software: "Comm Links" and "Accounting". The figure shows the Netscape

Navigator browser with four separate frames (characteristic of the SCN software). Each frame contains information about the different functions of the SCN software. The main menu is displayed in the upper-left frame. As links in the main menu are selected they open the sub menus in the remaining three frames. The "Comm Links" are shown in the lower right frame and "accounting" in the upper right frame. The "Comm Links" frame displays the interfaces that are configured on the host system and their current status. The "Accounting" frame provides information on transfer-rates, costs and times that the links were used based on account information. The lower left frame displays an expanded "Comm Link" menu used for configuration of specific interfaces.


Netscape: SeaNet
File Edit View Go Communicator Help
Back Forward Reload Home Search Guide Print Security Stop
Bookmarks Location
Internet Lookup New&Cool

http://131.120.63.227/SeaNet/SCN/

SeaNet – SCN v2.0.5 (SEANET GXA)

Clear All	Setup	Accounting
Comm Links	Scheduler	Queues
DataPipes	View Logs	Diagnostics
Utilities	Help	Partners
SLEEP		

Operator Log / Trouble Report /

SCN Comm Links

Link	Setup	Operations	Status
Ethernet	Do It	Do It	Do It
InMarSatB	Do It	Do It	Do It
Cellular	Do It	Do It	Do It
RF Link	Do It	Do It	Do It

All SCN Links

- Status

Show Network Status, Network Config

This page and all contents, are Copyright (C) 1994 by SeaNet Project

Current Accounting Rates – Refresh Display

Link Type	Rate Plan	Unique ID	Cost	Cost Unit	Start Period	End Period	Est. Thruput bytes/sec.
BHSD	OffPeak2	O2	10.00	min.	00:00:00	04:00:00	4000
BHSD	Peak	P1	10.00	min.	04:00:01	22:00:00	4000
BHSD	OffPeak1	O1	10.00	min.	22:00:01	23:59:59	4000
CELLULAR	All	C1	.50	min.	00:00:00	23:59:59	3600
MSAT	All	M1	1.00	min.	00:00:00	23:59:59	600

Link Status – Refresh Display

Net Interface	Address	State	RX Packets	RX Err/Drop	TX Packets	TX Err/Drop	Status
lo	127.0.0.1	UP	381	0/0	381	0/0	UP BROADCAST LOOPBACK RUNNING MTU 3584 Metric 1
ppp0	131.120.50.171	UP	415	0/0	539	0/0	UP POINTOPOINT RUNNING MTU 1500 Metric 1
eth0	131.120.63.227	UP	0	0/0	6	0/0	UP BROADCAST RUNNING MULTICAST MTU 1500 Metric 1

Show Network Status, Network Config

Figure 4.1 SeaNet Communication Node interface running via Netscape Navigator for the Linux operating system

C. NPS SEANET LABORATORY EQUIPMENT

The NPS SeaNet laboratory is divided into two sections, shipboard and shore-based. The shipboard section represents equipment that would normally be found on board a maritime vessel. This includes the KVH Tracphone and an IBM-compatible Personal Computer running the SeaNet Communications Software(SCN) software. The shore-based section includes a HP-UNIX workstation running OpenView and an Internet Service Provider (ISP) connection. Typical communication is via satellite link.

Although the laboratory's division indicates a shipboard section, all equipment is located at NPS. The shipboard and shore-based sections are named as to reflect their effective functionality. By maintaining all laboratory equipment in one area, researchers are able to manipulate either side of the laboratory and see the resulting effects on the opposite side. Ultimately, this laboratory is used to test and validate networking concepts before they are actually deployed on ships.

1. Shipboard Equipment

a. SeaNet Communications Node (SCN) System

The SeaNet Communications Node (SCN) System, also known as the shipboard computer, is essentially a Personal Computer hosting the SCN software. The primary functions of the NPS lab's shipboard computer is to communicate with the satellite telephone, run the SCN software, and to host an SNMP agent. The shipboard PC is configured with two operating systems, Linux and Microsoft Windows 95. Although actual shipboard computers use only the Linux operating system, this laboratory uses Window 95 as a comparison and setup tool. Justification for Window 95 is that the KVH Tracphone manual only references Windows and Macintosh data connections. Also, operator familiarity with Windows 95 allows researchers to evaluate functions in Windows 95 and then execute them in Linux.

Specifications for the NPS shipboard computer are as follows:

- Model: Dell OptiPlex GXa
- Processor: Intel Pentium II 233 MHz MMXI
- RAM: 64 MB EDO
- CDROM: Toshiba XM-6202B ATAPI 32X
- Operating System: Win95 / Red Hat Linux 5.0
- NIC: 3COM Fast Etherlink XL 10/100Mb TX Ethernet Adapter
- Monitor: Dell D1025TM.

b. KVH Tracphone Mobile Satellite Telephone System

The KVH Tracphone is used to establish a satellite data connection between the shore-based NMS and the shipboard PC. This equipment is provided by the American Mobile Satellite Corporation (AMSC) to allow research and evaluation of their satellite data circuit. There are three primary components to the KVH Tracphone: the antenna, the transceiver unit and the handset. The equipment is divided into above-deck and below-deck sections. The antenna is in the above-deck section while the transceiver and handset are in the below-deck section.

Figure 4.2 shows a basic schematic for the KVH Tracphone model used in the NPS SeaNet lab.

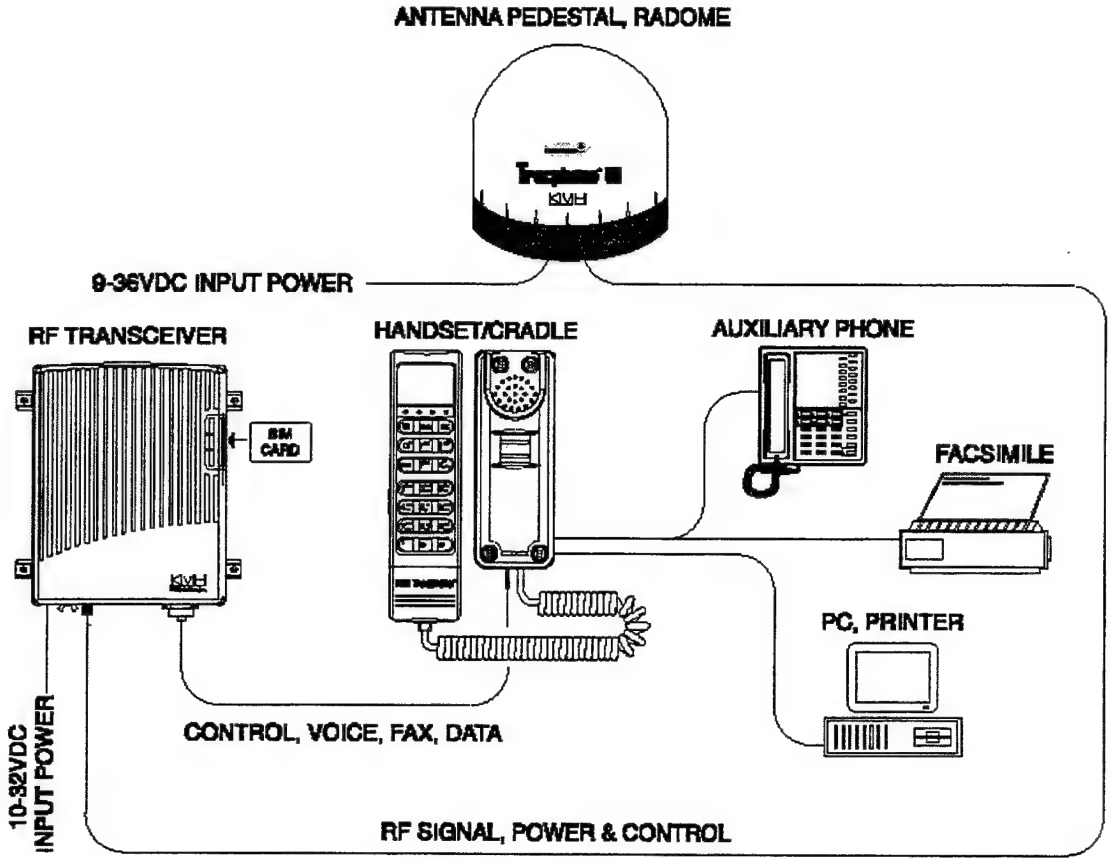


Figure 4.2 The KVH Tracphone [Ref. 15]

The KVH Tracphone is used to establish a data connection through the American Mobile Satellite Corporation owned satellite, AMSC-1. This geosynchronous satellite provides for voice and data communications. Currently AMSC supports a 4800 bps digital data channel through AMSC-1. As can be seen from Figure 4.3, the AMSC-1 satellite provides substantial coverage for vessels operating in North American coastal regions.

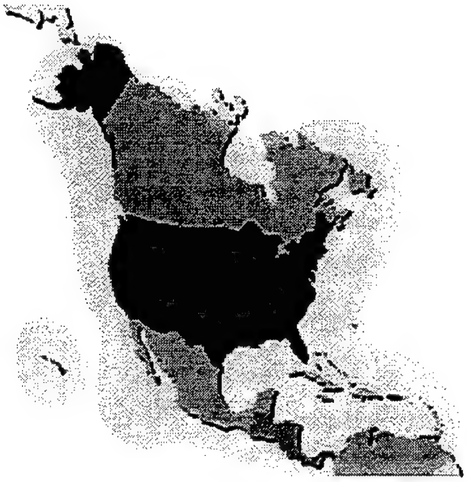


Figure 4.3 AMSC Satellite Coverage [Ref. 16]

2. Shore-Based Equipment

a. HP OpenView Workstation

The HP OpenView system is an extremely powerful network management tool capable of effectively monitoring a network of several thousand nodes. The NPS SeaNet laboratory, however, utilizes this tool to manage only the SeaNet Communications Node (SCN) system.

While HPOV represents an entire family of products, the NPS SeaNet laboratory utilizes only the HPOV Network Node Manager (NNM). NNM provides the following capabilities [Ref. 17]:

- View the current topology of your network as automatically discovered
- Easily diagnose network faults and performance problems conveniently from one location. This includes customizing and automating the monitoring of the ship's network and the management station's response to events.
- Access network and system configuration information for the nodes on your network without searching through files.
- Customize the management station by integrating existing applications into the HPOV windows menu bar.
- Plan for future networking needs. This includes collecting and storing MIB data for trend analysis.

The NPS laboratory's HPOV system hardware is shown in Figure 4.4

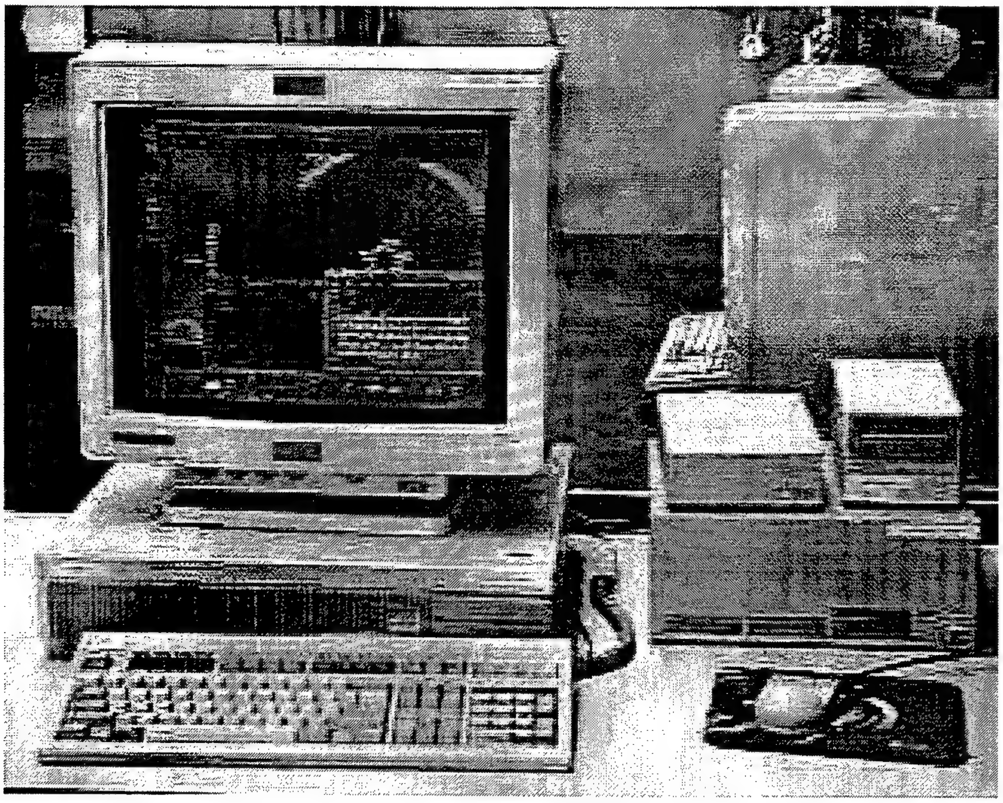


Figure 4.4 The NPS SeaNet Laboratory's HP OpenView Network Management Station, an HP Apollo Series 700 workstation running the HP-UX 10.01 operating system.

b. Internet Service Provider

One of the primary goals of the SeaNet program is to establish an Internet Service Provider (ISP) connection to maritime vessels. Currently the commercial company Omnet, a SeaNet partner, is fulfilling these requirements and expects to expand their operation as the number of SeaNet units increases. The primary ISP for the SeaNet lab is NPS. NPS hosts approximately 50 28.8 kbps modems. This bank of modems is used for connections between the shipboard computer and the NPS campus.

One of the major advantages of using NPS as an ISP is that the dial-in connection is contained within the firewalls of NPS. By containing all communications within NPS, the agent and NMS are able to exchange UDP packets without addressing firewall configurations or restrictions.

D. SETUP AND CONFIGURATION

Modem and point-to-point connection setup procedures for Windows 95 and Linux are detailed in Appendices A and B respectively.

E. OPERATION

The shipboard PC, regardless of operating system, issues standard AT style modem commands through a serial connection to the KVH transceiver unit. The KVH transceiver unit, in turn, issues commands to the antenna. The KVH antennae communicates with the AMSC-1 satellite over a specific line (e.g. voice, 2400bps or 4800bps). The satellite transmits to the AMSC downlink station in Reston ,Virginia and then completes the connection to the shore-based NMS via standard telephone land lines.

The shipboard and shore-based segments of the laboratory together allow the establishment of a 4800 bps point-to point connection between the HP workstation NMS and the SeaNet PC via the satellite. This connection can then be monitored and appropriate network management and other information can be transferred. Figure 4.5 illustrates a typical connection using the KVH Tracphone. The connection is initiated from the shipboard side with the SCN system. The initial modem commands come from Windows 95 or Linux and are sent to the KVH TU.

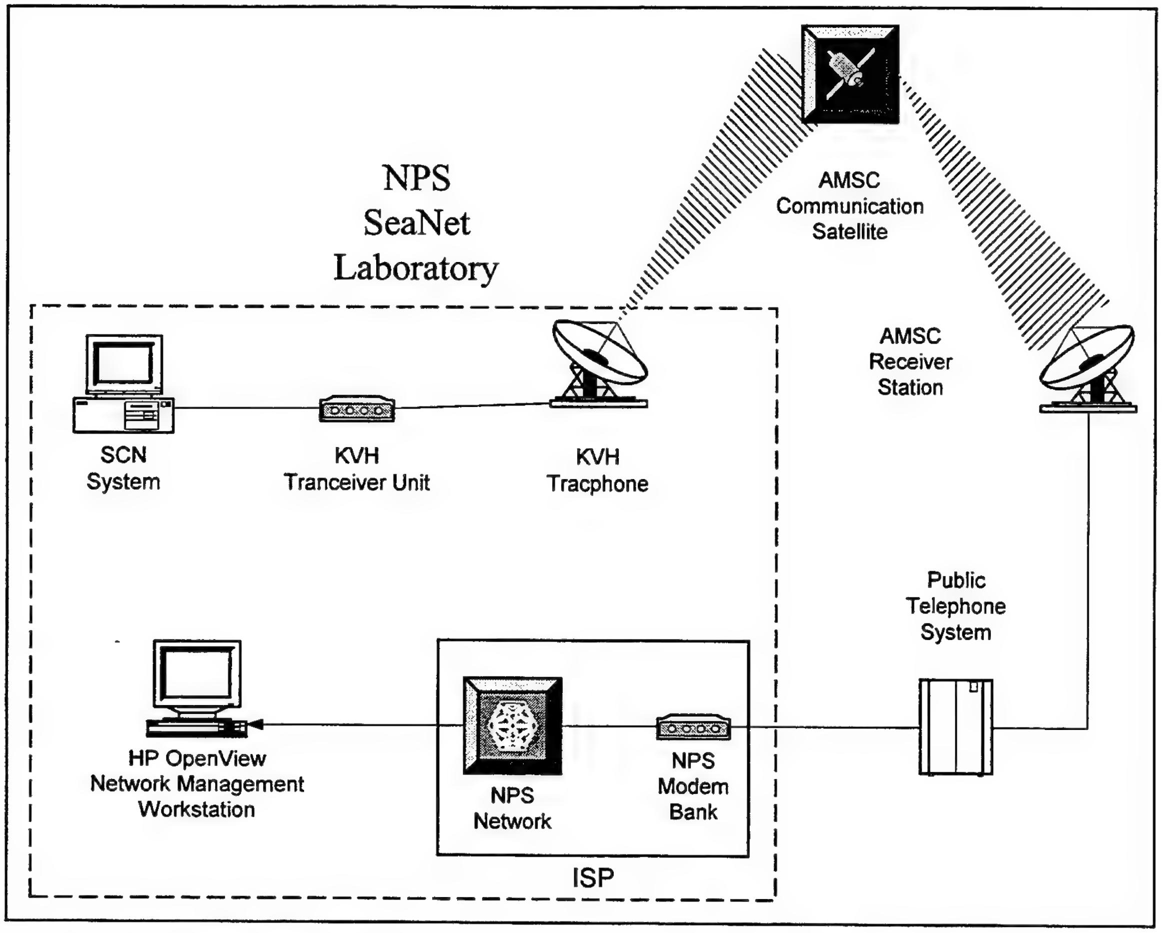


Figure 4.5 The NPS SeaNet Laboratory, a typical connection between the network management station and the SeaNet Communication Node (SCN) system.

The TU processes the commands and initiates a call to the ISP being dialed. The SCN system establishes a connection with an ISP through the satellite using the KVH TU as a modem. After a connection is established then the connection behaves like any other dial-up connection. This allows for access to the Internet at 4800bps. While this is low-bandwidth compared with regular modem and LAN connections, it is capable of supporting e-mail and other low data-rate functions such as selected experimental data transfer.

F. SUMMARY

The SeaNet laboratory was developed to test the potential for remote network management of shipboard computers. The shipboard side of the SeaNet laboratory consists of a shipboard computer hosting a Linux SNMP agent and a KVH Tracphone. The shore-based side consist of an HP workstation hosing HPOV's NNM. These two divisions are connected by dialing into the NPS modem bank using a 4800 bps satellite data connection.

V. EXPERIMENTAL RESULTS

A. INTRODUCTION

This chapter discusses research findings. First, it addresses physical connection issues (such as latency and data rate) encountered with the AMSC Tracphone and AMSC satellite connection. Second, it addresses the ability of HP OpenView to manage the NPS simulated shipboard node. Finally, it considers problems encountered in attempting to manage a remote node with a dynamically assigned IP address.

B. AMSC TRACPHONE FINDINGS

This section discusses the physical properties of the satellite connection including the latency and throughput observed at NPS. A brief discussion about the Tracphone's ability to switch between voice and data communications is also included.

1. Latency and Throughput

Latency was observed by using the `ping` command to ping the shipboard computer from the network management station. The average latency observed was approximately 2.2 seconds over 1000 pings. No pings took less than 2.1 seconds or more than 2.4 seconds.

Throughput of the connection was measured by using the File Transfer Protocol to transfer files between computers. Files were transferred to the shipboard computer from either the NPS shore-based computer (HP) or the file server at WHOI. The shipboard computer is denoted by the operating system it was running at the time of file transfer: Linux or Windows 95. Two sources for dial-in connection were used: NPS and Redshift (http://www.redshift.com) , a local Internet Service Provider (ISP). The data rates encountered are found in Table 5.1.

From	To	File Size (bytes)	Time (min)	Data Rate (Kbytes/sec)	Data Rate (Kbytes/min)	SNMP Monitored?
WHOI	Linux	324,391	32	0.17	10.2	Yes
WHOI	Linux	324,391	28.8	0.22	13.2	Yes
WHOI	Linux	324,391	18.8	0.28	16.8	Yes
HP	Linux	427706	24	0.29	17.4	Yes
HP	Linux	427,706	20	0.35	21	Yes
Win95	HP	427,706	16.5	0.43	25.8	Yes
HP	Win95	116,614	5.4	0.36	21.6	Yes
Win95	HP	106,614	9.2	0.19	11.4	Yes
WHOI	Win95	398,243	16.6	0.40	24	No
WHOI	Win95	398,243	15.6	0.43	25.8	No
WHOI	Win95	398,243	15.2	0.55	33.0	No

Table 5.1 Data Transfer Rates

Conducting continuous network management (Section D) while performing other functions like file transfer will have some impact on the data transfer rate. To illustrate this impact, consider the following exercise: An SNMP request/response query is approximately 82 bytes or 656 bits [Ref 13]. Considering that a conservative estimate of the reply/response cycle over the satellite link is 2.4 seconds, potentially 273 bps may be consumed by network management. This number represents roughly six percent of the available bandwidth provided by a 4800 bps connection. Despite this estimation, the values of Table 5.1 provide no direct correlation between the rates of managed and unmanaged data transfers. Obviously Table 5.1 presents only a small sample of data transfer rates and these rates are affected by several factors other than network management (server loads, network traffic, etc.). Clearly, a more rigorous investigation of the overhead associated with constant network monitoring is warranted.

2. Smooth Transition Between Voice and Data Communications

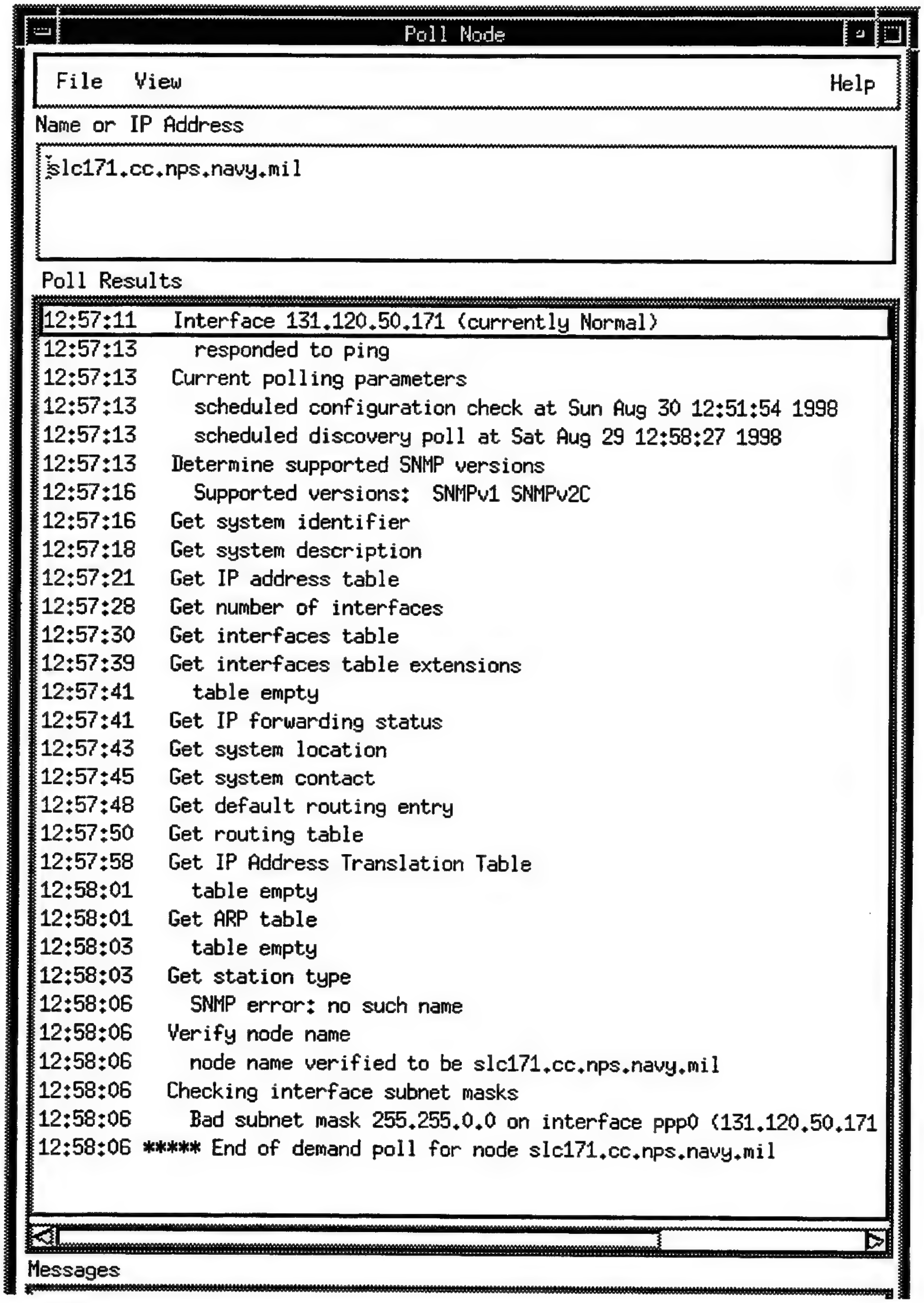
One notable feature of the AMSC Tracphone is its ability to easily switch between data and voice communications. Although voice and data cannot be transmitted simultaneously, the operator can switch to voice operations simply by using the Tracphone handset: pushing the "END" button on the handset terminates data operations and commences voice communications. For a ship with minor data transfer requirements the AMSC Tracphone can serve as a backup voice communications system.

C. HPOV NETWORK MONITORING

This section discusses the information HPOV is able to extract from the shipboard SNMP agent. A listing of all MIB-II variables and instances extracted from the shipboard computer is given in Appendix D.

1. Poll Node

One of the most powerful and fundamental functions HPOV offers is that of Poll Node. The Poll Node command can be initiated from the toolbar by selecting "Fault" and then "Network Configuration: Poll Node." Using the Poll Node command, a network administrator can direct the NMS to update all available information on a given SNMP agent. Poll Node surveys network devices for their SNMP capabilities, an agent's polling parameters, and which SNMP applications the agent can support. Figures 5.1 displays the results of a Poll Node command performed on a LINUX agent. Note that from Figure 5.1 one can observe that Shipboard Linux computer supports both SNMPv1 and SNMPv2C.



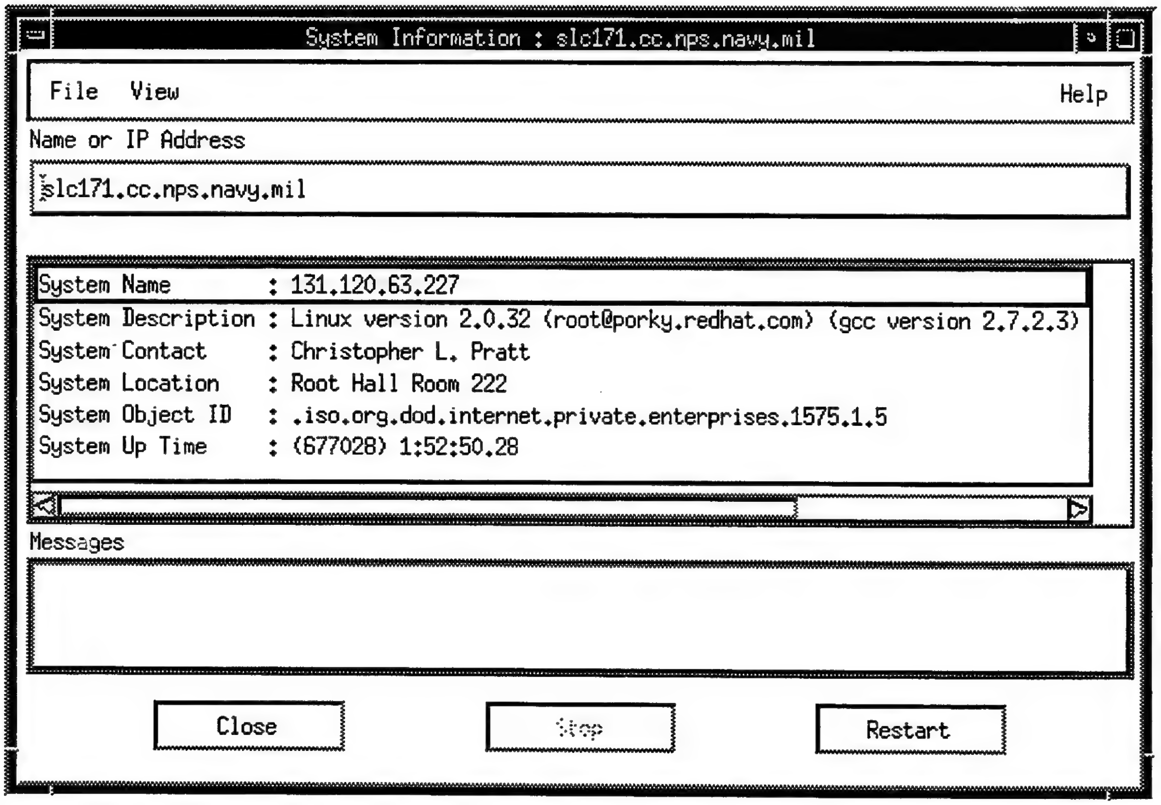
Name or IP Address: slc171.cc.nps.navy.mil

```
12:57:11   Interface 131.120.50.171 (currently Normal)
12:57:13     responded to ping
12:57:13   Current polling parameters
12:57:13     scheduled configuration check at Sun Aug 30 12:51:54 1998
12:57:13     scheduled discovery poll at Sat Aug 29 12:58:27 1998
12:57:13   Determine supported SNMP versions
12:57:16     Supported versions:  SNMPv1 SNMPv2C
12:57:16   Get system identifier
12:57:18   Get system description
12:57:21   Get IP address table
12:57:28   Get number of interfaces
12:57:30   Get interfaces table
12:57:39   Get interfaces table extensions
12:57:41     table empty
12:57:41   Get IP forwarding status
12:57:43   Get system location
12:57:45   Get system contact
12:57:48   Get default routing entry
12:57:50   Get routing table
12:57:58   Get IP Address Translation Table
12:58:01     table empty
12:58:01   Get ARP table
12:58:03     table empty
12:58:03   Get station type
12:58:06     SNMP error: no such name
12:58:06   Verify node name
12:58:06     node name verified to be slc171.cc.nps.navy.mil
12:58:06   Checking interface subnet masks
12:58:06     Bad subnet mask 255.255.0.0 on interface ppp0 (131.120.50.171
12:58:06 ***** End of demand poll for node slc171.cc.nps.navy.mil
```

Figure 5.1 The HPOV Poll Node Table showing the process of the NMS retrieving data from the SNMP agent located at IP address 131.120.50.171.

2. System Information

System information is another standard menu selection provided by HPOV. This information can be found by selecting "Configuration" in the tool bar, then "System Information." Representative System Information results are shown in Figure 5.2.



Name or IP Address

slc171.cc.nps.navy.mil

```
System Name        : 131.120.63.227
System Description : Linux version 2.0.32 (root@porky.redhat.com) (gcc version 2.7.2.3)
System Contact     : Christopher L. Pratt
System Location    : Root Hall Room 222
System Object ID   : .iso.org.dod.internet.private.enterprises.1575.1.5
System Up Time     : (677028) 1:52:50.28
```

Figure 5.2 The HPOV System Information Table

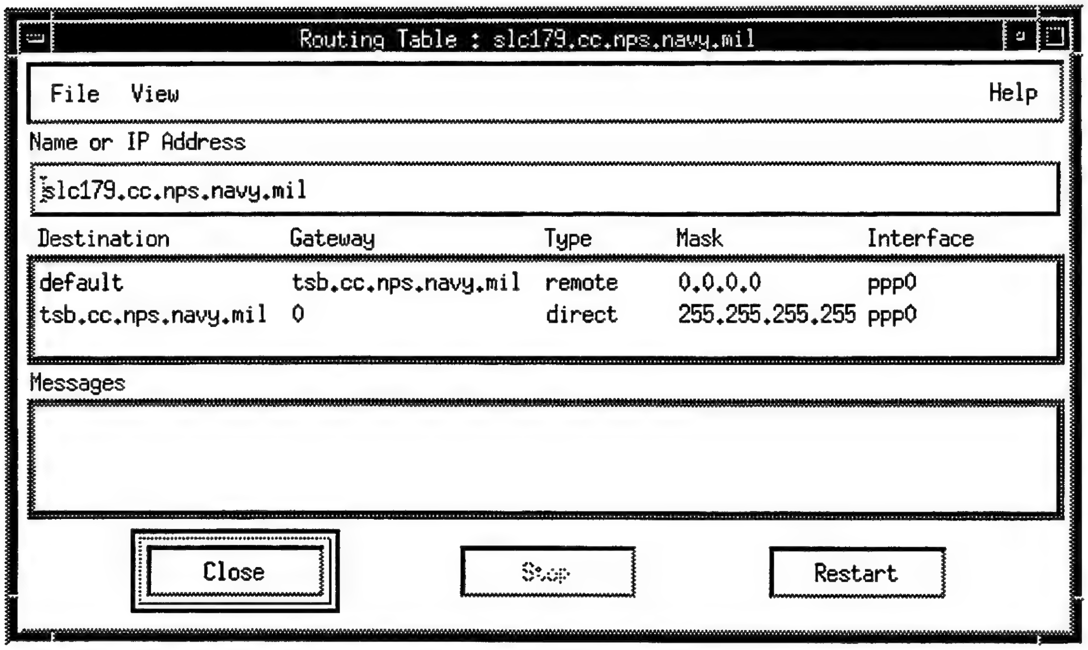
As can be seen from Figure 5.2, the System Information command returns six instances from the agent. The information found in system information is a representation of the MIB-II `systems` group. *System Name* denotes the administratively assigned name of the SNMP agent. The *System Description* field denotes the operating system of agent computer. The *System Contact* and *System Location* fields identify who to contact about the system and where the system is located respectively. The *System Object ID* represents Red Hat's authoritative network management subsystem. *System Up Time* denotes the elapsed time since the system was last re-initialized.

3. Network Configuration

Another standard service provided by HPOV is Network Configuration. Network Configuration is broken into three sections: Address Resolution Protocol (ARP) Cache, Routing Table, and Services. In the case of the shipboard computer, the ARP Cache Table contains no information.

a. Routing Table

The routing table command provides information about possible destinations and how to reach them [Ref. 13]. This information is useful for diagnosing connectivity problems [Ref. 17]. From Figure 5.3 one can observe that the default destination gateway is tsb.cc.nps.navy.mil. This default gateway is the route the SNMP agent will use when it cannot find a specific route. The destination can be one of three types: direct (directly connected to a LAN), invalid (not currently available) or remote (through a remote gateway). Interface indicated the name of the interface used to reach the destination [Ref. 17].



Name or IP Address

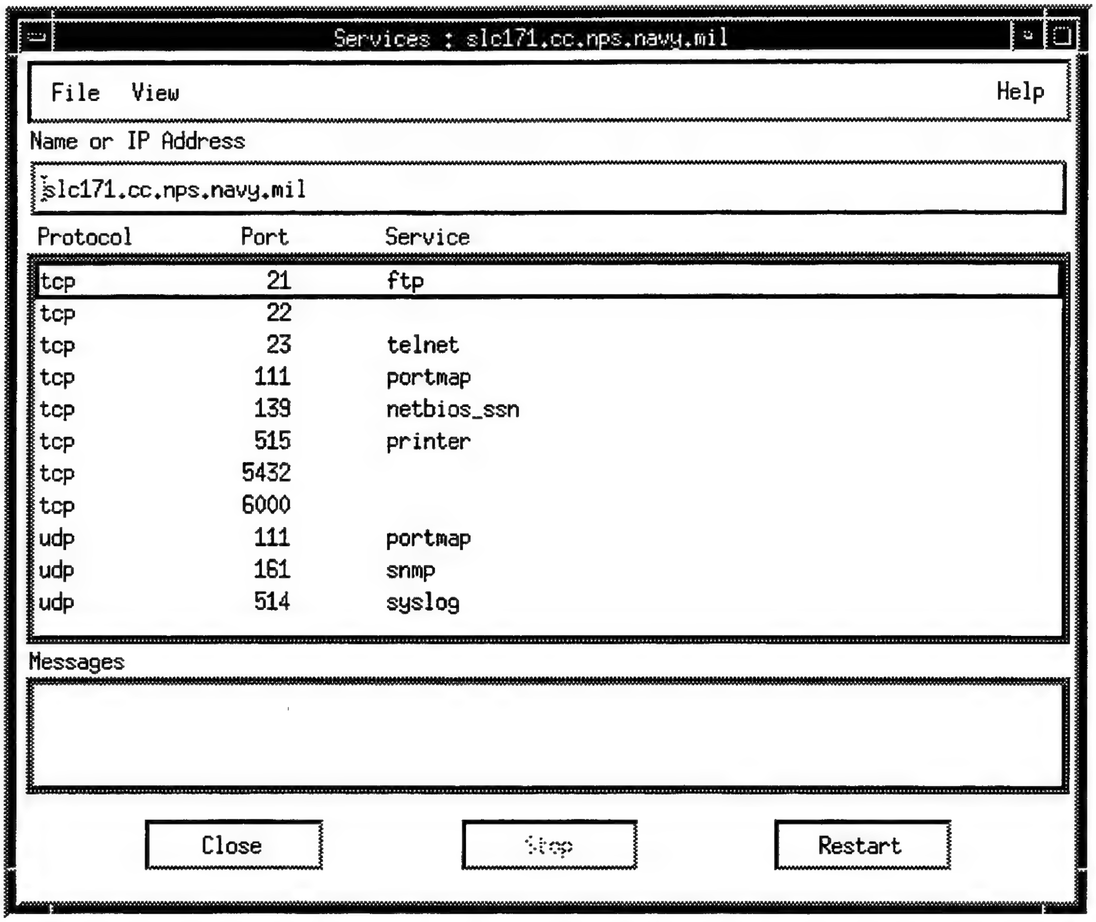
slc179.cc.nps.navy.mil

```
Destination            Gateway                 Type      Mask               Interface
default                tsb.cc.nps.navy.mil     remote    0.0.0.0            ppp0
tsb.cc.nps.navy.mil    0                       direct    255.255.255.255    ppp0
```

Figure 5.3 The HPOV Routing Table

b. System Services

The System Services table provides information on port usage by the agent computer. "This operation is useful for determining what services a node is configured to support" [Ref. 17]. In Figure 5.4, one can observe that eight TCP ports and three UDP ports are in use. These port numbers correlate directly to processes running on the agent computer. The service column of the table lists the service for which the node is listening (if known).



Name or IP Address: slc171.cc.nps.navy.mil

Protocol	Port	Service
tcp	21	ftp
tcp	22	
tcp	23	telnet
tcp	111	portmap
tcp	139	netbios_ssn
tcp	515	printer
tcp	5432	
tcp	6000	
udp	111	portmap
udp	161	snmp
udp	514	syslog

Figure 5.4 The HPOV Services Table

D. MONITORING FILE TRANFERS

This section illustrates the real-time monitoring of an SNMP agent from a remote network management station. During a file transfer, the HPOV network node manager is observing IP packet and TCP segment information. In this example, a 324 Kbytes file is being transferred from a WHOI file server to the NPS shipboard computer. Figures 5.5 and 5.6 show the SNMP monitoring of this file transfer from the HPOV system. Using this functionality, a shore-based network administrator can monitor ship-to-shore data transfers in real time.

Six variables are monitored in these graphs. The variables are as follows:

- ipInReceives (MIB 1.3.6.1.2.4.3): A counter which gives the total number of input datagrams received from interfaces, including those received in error [Ref. 18].

- ipInHdrErrors (MIB 1.3.6.1.2.4.4): A counter which gives the number of input datagrams discarded due to errors in IP header [Ref. 18].

- ipInAddrErrors (MIB 1.3.6.1.2.4.5): A counter which gives the number of input datagrams discarded because the IP address in the destination field was not valid to be received at this entity [Ref. 18].

- ipInDiscards (MIB 1.3.6.1.2.4.8): A counter which gives the number of input IP datagrams for which no problems were encountered to prevent their continued processing but which were discarded [Ref. 18].

- tcpInSegs (MIB 1.3.6.1.2.6.10): A counter which provides the total number of segment received, including those received in error [Ref. 18].

- tcpRetransSegs (MIB 1.3.6.1.2.6.12): A counter which gives the total number of retransmitted segments [Ref. 18].

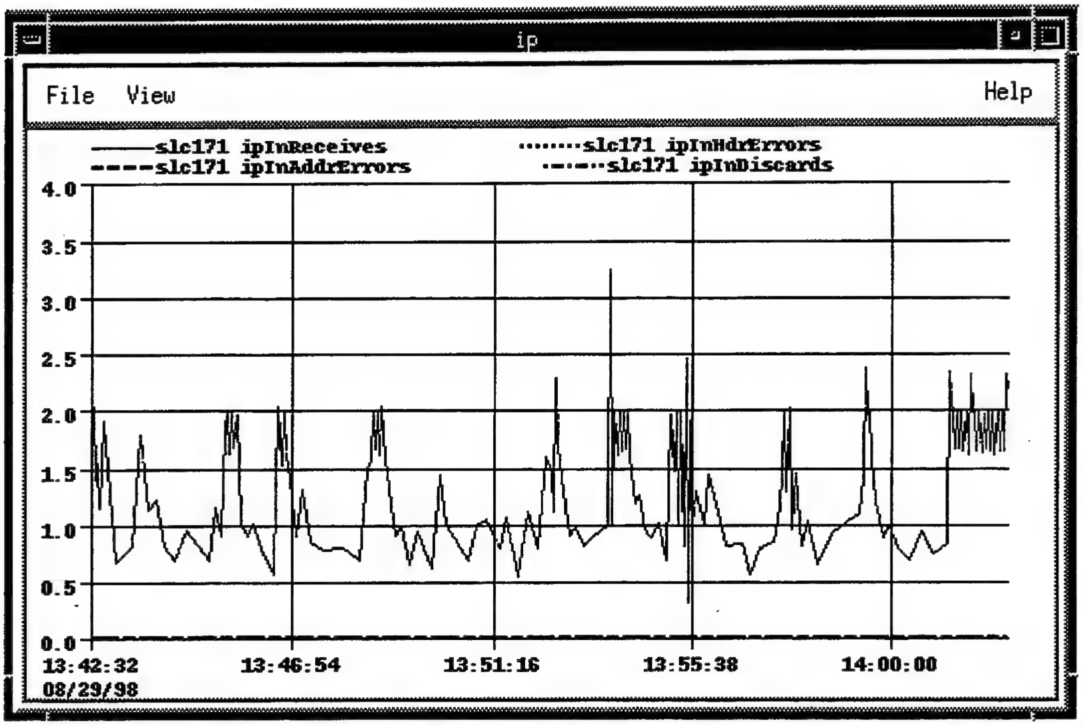


Figure 5.5 IP Packet Monitoring of File Transfer from WHOI to Linux Agent

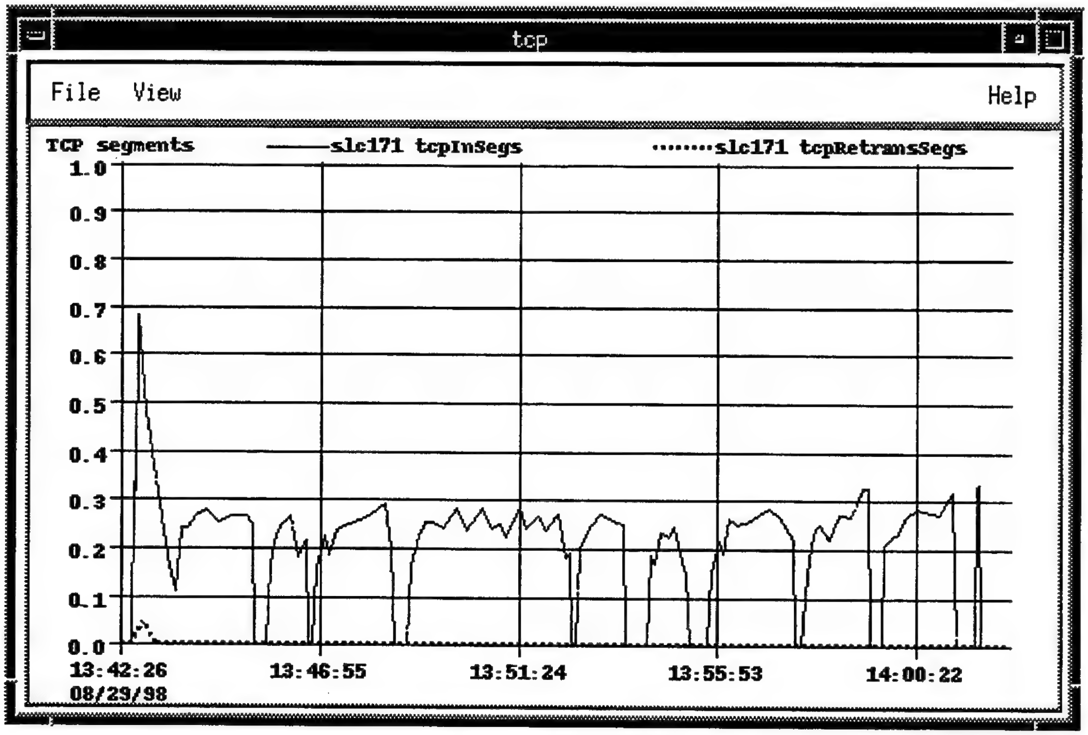


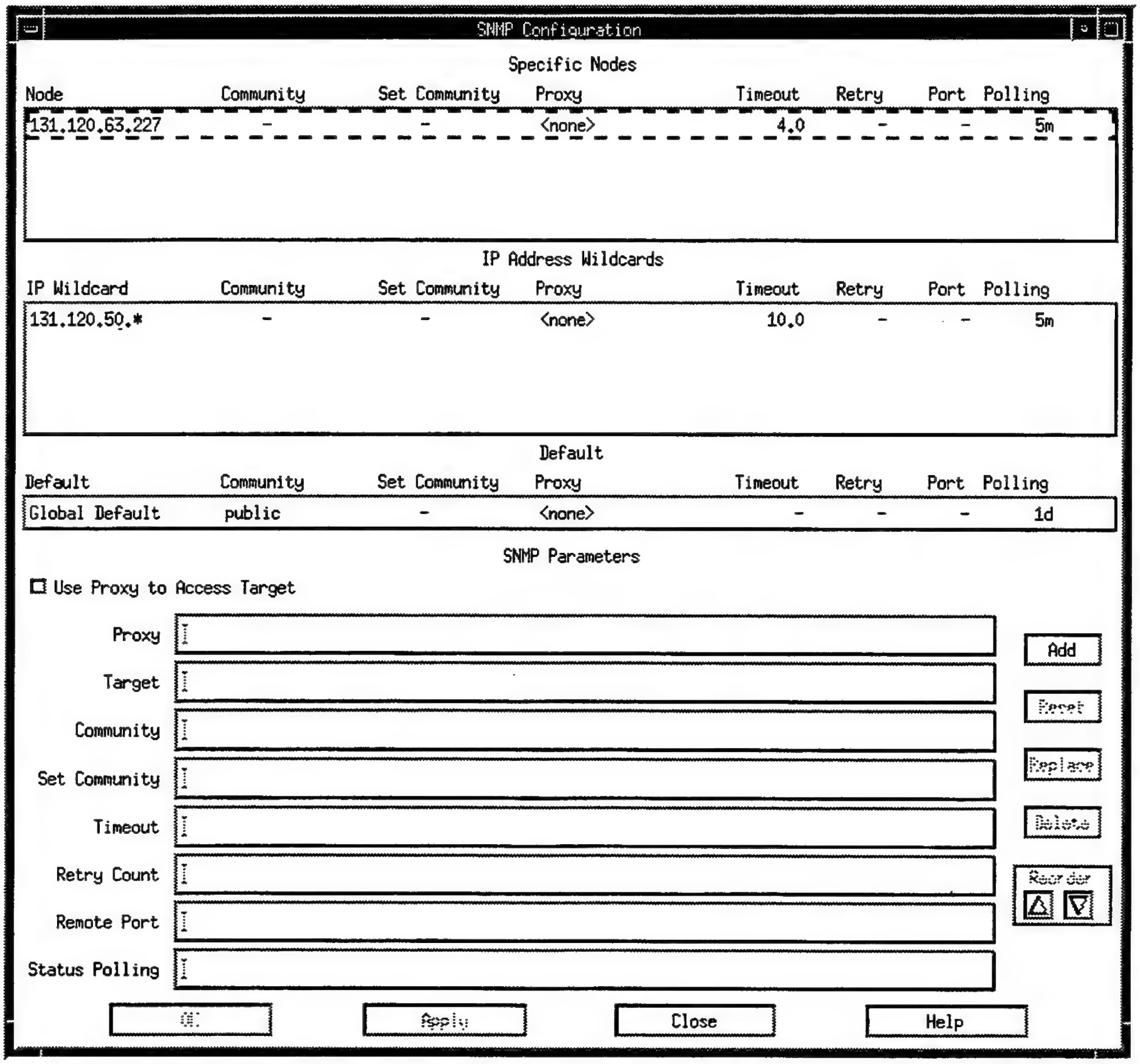
Figure 5.6 TCP Segment Monitoring of File Transfer from WHOI to Linux Agent, note that the initial `tcpRetransSegs` are probably TCP retransmissions stacking up as they time out. Initial TCP timeouts may be less than the propagation time through the satellite link while the timeouts of later packets are adjusted to be larger than the satellite link delay time.

E. PROBLEMS

1. Latency and Timeouts

The data response latency of roughly 2.5 seconds can lead to SNMP request time-outs. Time-outs represents the amounts of time (in seconds) that HPOV will wait for a response before retrying the SNMP request. If HPOV encounters three consecutive timeouts, it will suspend polling of the node. Suspension of polling can be avoided by setting the HPOV polling interval to a significantly long interval. Selecting "Options" then "Configure Node Status Polling Intervals" sets node polling intervals.

Figure 5.7 shows the HPOV SNMP Configuration window. Note that the entry 131.120.50.* has a time-out of 10 seconds. This wildcard value represents all nodes in the NPS modem bank. Such a configuration prevents the time-out of the shipboard SNMP agent regardless of which IP it is dynamically assigned.



SNMP Configuration

Specific Nodes

Node	Community	Set Community	Proxy	Timeout	Retry	Port	Polling
131.120.63.227	-	-	<none>	4.0	-	-	5m

IP Address Wildcards

IP Wildcard	Community	Set Community	Proxy	Timeout	Retry	Port	Polling
131.120.50.*	-	-	<none>	10.0	-	-	5m

Default

Default	Community	Set Community	Proxy	Timeout	Retry	Port	Polling
Global Default	public	-	<none>	-	-	-	1d

SNMP Parameters

□ Use Proxy to Access Target

Proxy

Target

Community

Set Community

Timeout

Retry Count

Remote Port

Status Polling

Figure 5.7 The HPOV SNMP Configuration Window

2. Dynamically Assigned IP Addresses

Dynamically assigned IP addresses present another problem for network management. HPOV is designed to monitor a stable network; that is, a network with fixed IP addresses and devices that are consistently on the network. The shipboard computer, however, is expected to dial in, receive a dynamic IP address, exchange data, and disconnect. Total connection time for the ship may be as little as thirty minutes. HPOV, however, may require several hours to discover a new network node.

To increase the frequency with which HPOV will discover new nodes (and increase the chance that HPOV will discover the shipboard computer), HPOV can be configured to rapidly poll nodes. Unfortunately, rapid polling has limitations as well. If the polling interval is too frequent, HPOV will never discover the node; a new polling cycle will begin before the prior polling iteration is complete. Thus some nodes are polled rapidly while others are not polled at all.

3. Node Merge

A unique problem for the NPS SeaNet laboratory is that the shipboard computer is also the NPS LAN node 131.120.63.227 (Figure 5.8). When the shipboard computer dials into the NPS modem bank (Figure 5.9), HPOV will discover that the two nodes (the modem bank node and the LAN node) represent the same computer. HPOV will then merge the nodes, representing them as a modem bank node in both the 131.120.63 and 131.120.50 subnetworks (Figures 5.10 and 5.11 respectively). Once the nodes are merged, HPOV recognizes that both are SNMP supported and SNMP information can be gathered from either node representation.

The problem is that after disconnecting from the NPS modem bank, the 131.120.63 node never relinquishes its status as a modem bank node; that is, once the node 131.120.63.227 is converted to slc170, it never reverts back to 131.120.63.227. It may take several hours (or even several days) for HPOV to realize that the modem bank node no longer represents the LAN node and rediscover the LAN node.

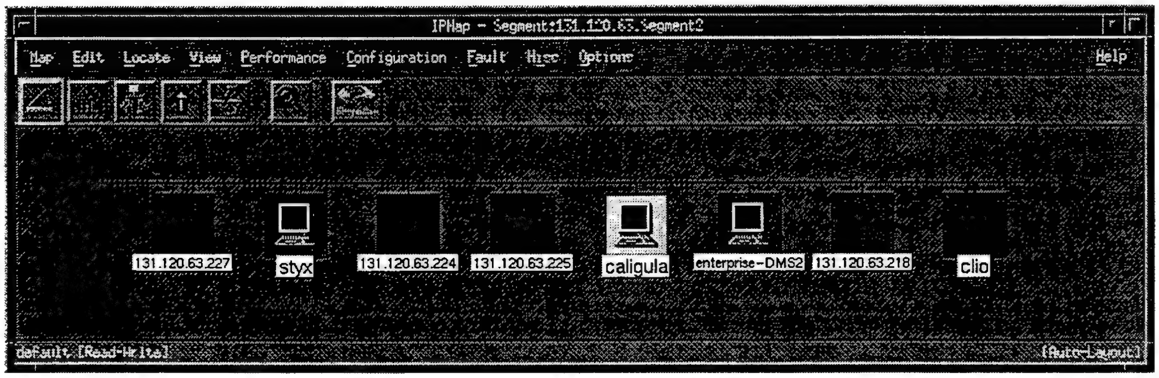


Figure 5.8 The 131.120.63 Subnet Prior to Node Merge

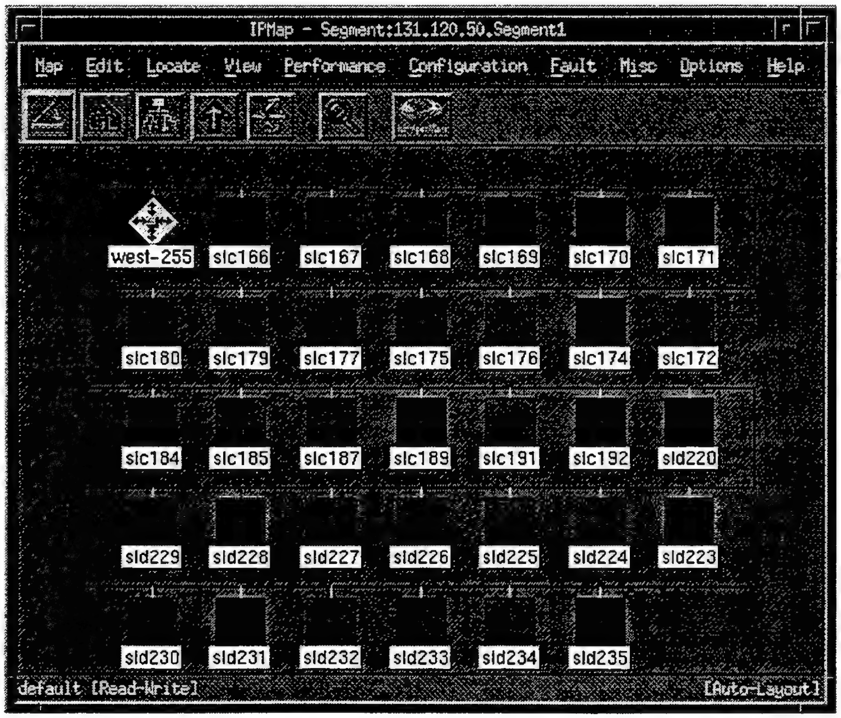


Figure 5.9 NPS Modem Bank Prior to Node Merge (Note node slc170)

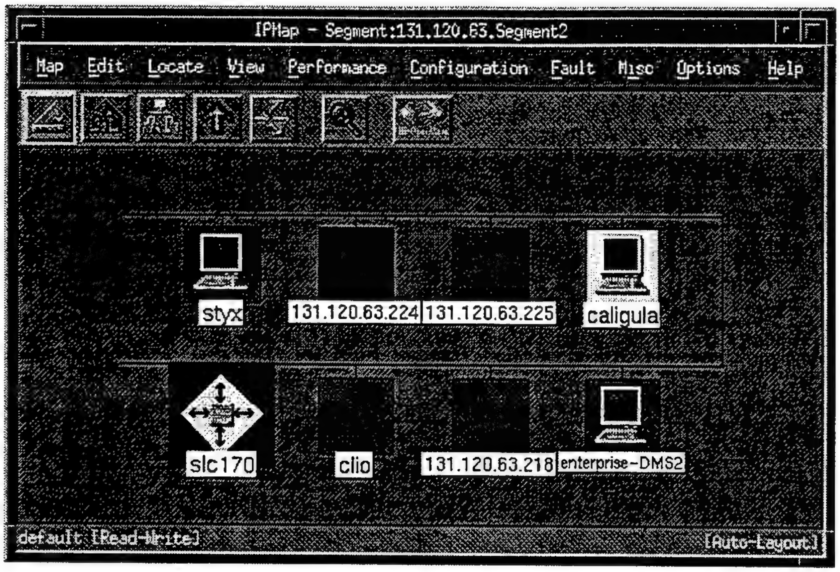


Figure 5.10 The 131.120.63 Subnet After Node Merge (Note that node 131.120.63.227 is now represented by slc170)

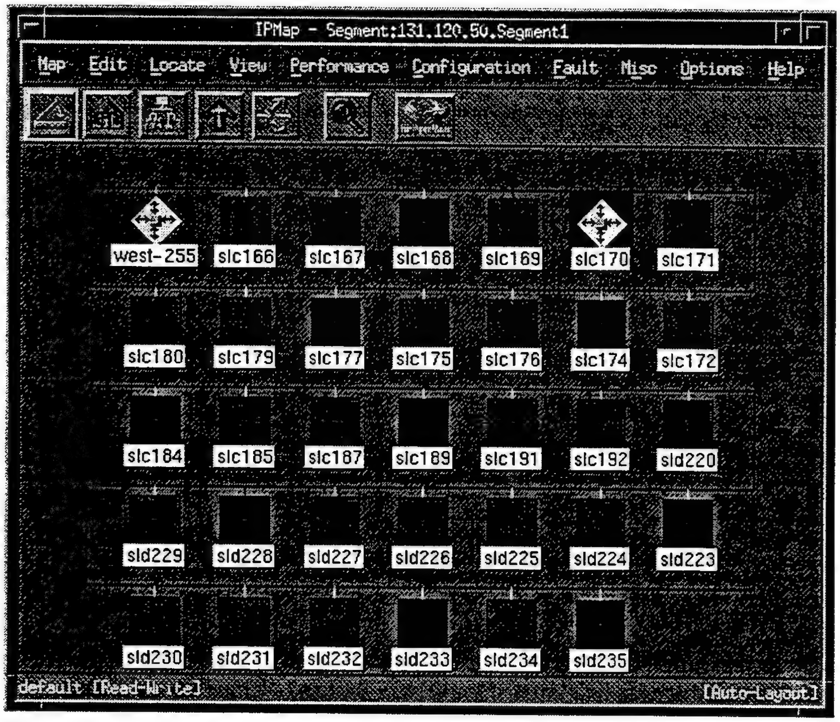


Figure 5.11 The NPS Modem Phone Bank After Node Merge

F. SUMMARY

HPOV is a powerful tool for remote network management. Inherent HPOV functions such as Poll Node, Routing Table, and System Services provide easy monitoring of SNMP agents. Real time monitoring of SNMP variables also provides a powerful tool for network administrators to gather SNMP information. HPOV does have some difficulties when monitoring the NPS shipboard computer, however. Latency imposed by satellite connections can lead to HPOV time-outs. HPOV also has difficulty in monitoring devices with dynamically assigned IP addresses. This difficulty is compounded by the relatively short connection times it is anticipated that ships will make. Further work is needed to resolve these difficulties.

VI. CONCLUSIONS AND RECOMMENDATIONS

A. RESEARCH CONCLUSIONS

This thesis investigates the concept of remote network management over a satellite link. It surveys the current SeaNet program, discusses the principles of network management, and introduces the NPS SeaNet laboratory. Finally, this thesis presents initial findings and problems associated with actual remote network management.

Maritime communications tend to be expensive and of limited capacity. The SeaNet program is building an infrastructure to provide Internet access to maritime vessels at a reasonable cost. One aspect of the SeaNet program is remote network management. Remote network management alleviates network administration burdens from shipboard personnel and allows them to concentrate on more mission specific functions. It also allows for dedicated monitoring and analysis from the shore-based management site. This capability will support the SeaNet program by providing the maritime research community with a flexible and cost-effective tool for monitoring sea-based assets.

The NPS SeaNet lab is investigating emerging technologies for the SeaNet program. Using an IBM compatible PC, HP-Unix workstation running HPOV, and a KVH Tracphone, the NPS SeaNet laboratory is able to establish a point-to-point connection via an AMSC satellite connection. Remote network management is possible, but it is not without difficulties. HPOV encounters difficulties when dealing with latency and dynamically assigned IP addresses.

B. RECOMMENDATIONS FOR FURTHER RESEARCH

1. Remote Network Monitoring

The next logical step for the NPS SeaNet laboratory involves investigating the use of RMON probes on shipboard network devices. An RMON probe can gather information on the shipboard LAN while the ship is disconnected from the shore-based network management station. When the ship conducts data transfers, condensed RMON data can be exported to the shore-based site for analysis.

2. Network Management of Actual Maritime Vessels

Another logical step for the NPS SeaNet laboratory involves the monitoring of actual shipboard SNMP agents. This task involves configuring the NMS to observe WHOI vessels through Omnet. When ships connect to the Omnet site, the NMS would be able to gather information from the vessels. This project would also include monitoring certain aspects of the Omnet shore station. WHOI and Omnet representatives have favorably received suggestions for this project.

3. Discovery of Dynamically Assigned IP Addresses

The difficulty HPOV encounters with dynamically assigned IP addresses has not been fully addressed. For HPOV to function as a NMS for agents with dynamically assigned IP addresses, it must acknowledge the agents when they first dial in to the network. This project would involve creating a way for the shipboard agent to send a message to the NMS as soon as the agent connected to the network. The NMS would, upon receipt of message, poll the IP address for SNMP information. This way all SNMP information could be gathered before shipboard computer disconnects from the network.

4. Customization of MIB variables for SeaNet SNMP agents

Research is needed to determine exactly what kind of information should be gathered from maritime vessels and how this information can be used. This project would involve the examination of MIB variables beyond the MIB-II subset. Study would include evaluation of MIBs developed by private enterprises and the possible creation of MIBs unique to the SeaNet project.

5. Evaluation of Future SeaNet Equipment

This study would involve evaluation of technologies beyond those currently being used in the SeaNet program. Current initiatives are underway for NPS's evaluation of a 56kbyte satellite link provided by AMSC.

6. Development of a Bandwidth Budget

This project would develop metrics for estimating the bandwidth requirement of a SeaNet vessel. It would involve studying current shipboard LANs, estimating the SNMP traffic across the LAN and SNMP traffic across the satellite connection. Questions to be answered include: what overhead is associated with remote network management and does this overhead detract from data transfer operations? This study would directly expand on the throughput and latency issues introduced in Chapter 5.

APPENDIX A. WINDOWS 95 CONFIGURATION

This appendix describes how to:

1. Setup a serial connection between the KVH Transceiver Unit (TU) and a PC using an RS-232 cable.

2. Use the Windows95 HyperTerminal to check the serial connection between the PC and the KVH Transceiver Unit (TU)

3. Setup Windows Dial-Up Networking to use the KVH Tracphone to dial an ISP.

4. Setting up a serial connection between a PC and the KVH Transceiver Unit(TU)

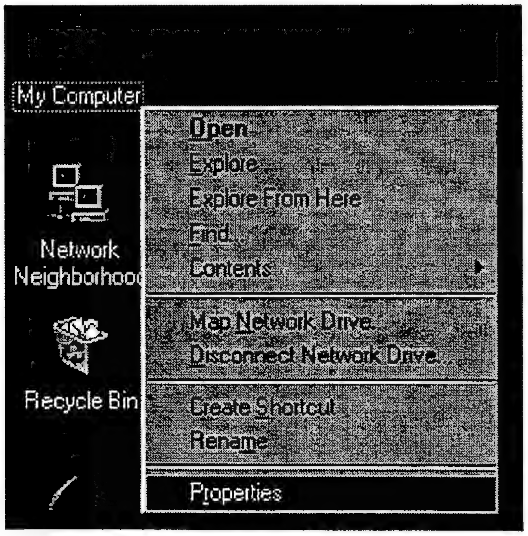
This section should be used in conjunction with the KVH Tracphone manual. The RS-232 is recommended to be 6 feet in length or LESS. The pin configuration of an RS-232 cable is DB9F to DB25M, which indicates a 9 pin female end and a 25 pin male end respectively. Before connecting the cable between the computer and the TU both the computer and the Tracphone must be off. The cable's 9-pin female end connects to the computer's 9 Pin male serial port.

NOTE! When connecting the cable, identify which computer serial port you are connecting the cable to i.e., COM1, COM2, etc. The port will be labeled on the back of the computer or in your computer documentation. This information will be needed later in the setup.

The cable's 25 pin male end connects to the KVH TU's 25 pin female port. This will establish a serial connection to the KVH TU. The KVH TU accepts "AT" commands like any standard modem. The KVH Tracphone manual recommends powering up the computer first and then the KVH Tracphone.

1. Power up the Computer
2. Power up the KVH Tracphone Antennae (Above deck equipment)
3. Power up the KVH Transceiver Unit (Below deck Equipment)

- RIGHT CLICK "My Computer" (click the right mouse button).



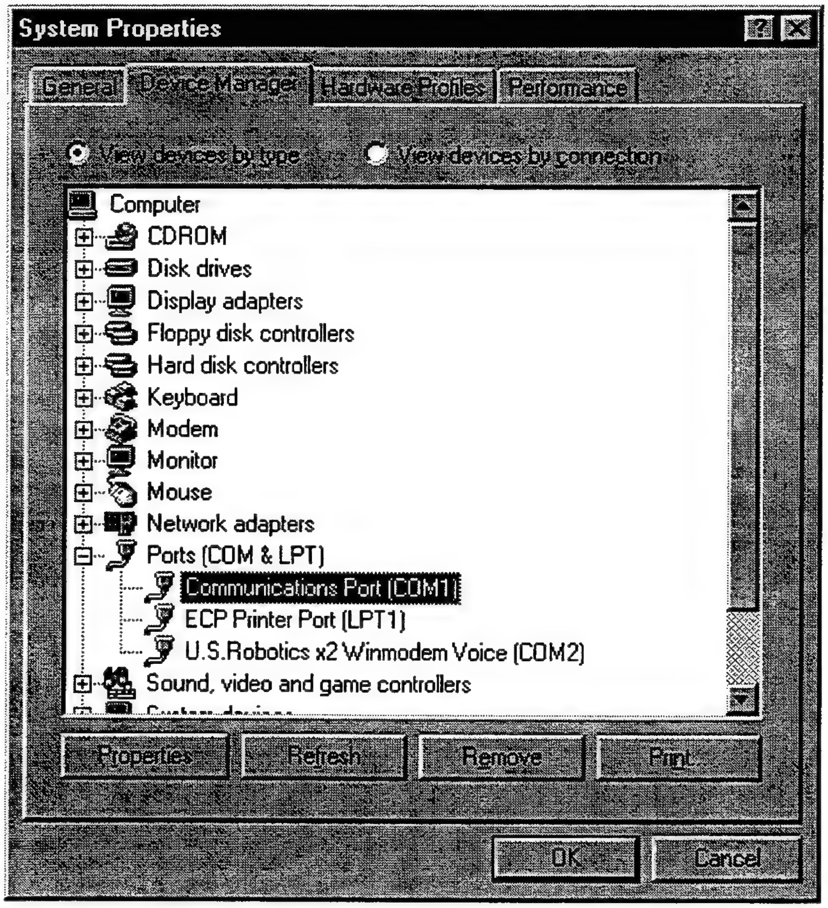
- LEFT CLICK (click the left mouse button) to open the "System Properties" dialog box.

- LEFT CLICK on the "Device Manager" tab in the "System Properties" Dialog box.

- LEFT CLICK the "+" symbol next to "Ports (COM & LPT)" to expand the "Ports (COM & LPT)" section.

- Identify the Communication Port that you have used to connect your computer to the KVH TU with an RS-232 cable. In our example, it is on "Communications Port (COM1)".

- LEFT CLICK on the correct Communications Port to HIGHLIGHT it.

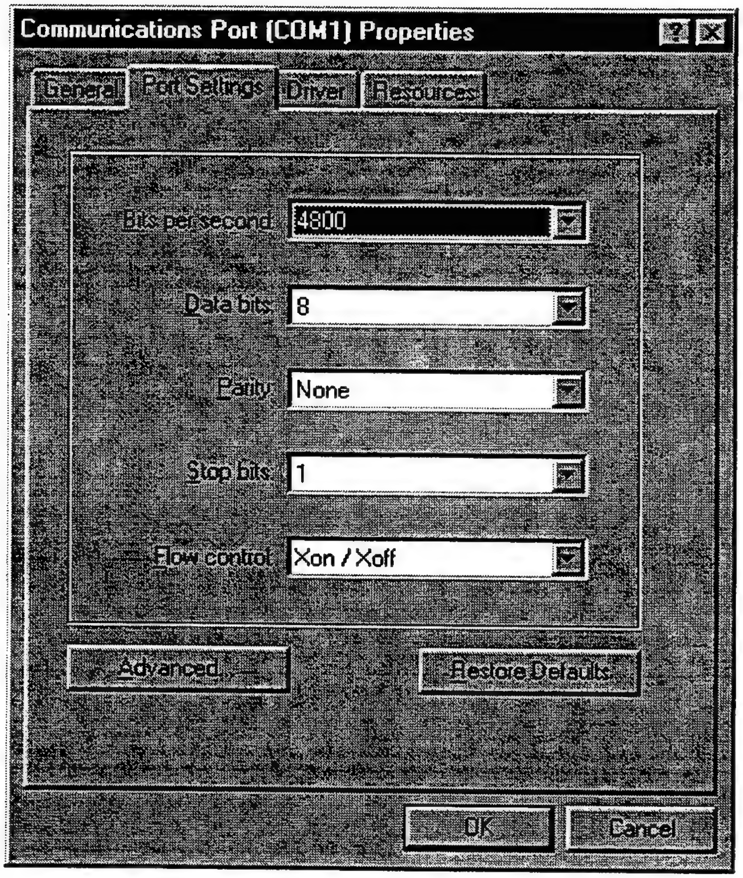


- LEFT CLICK the "Properties" button below the "Device Manager" window. This will open the "Communication Port (COM1) Properties" dialog box.

- LEFT CLICK the “Port Settings” tab on the “Communications Port (COM1) Properties” dialog box.

- SELECT the following values in the “Port Settings” drop down boxes (LEFT CLICK the drop down box arrow, HIGHLIGHT the desired value and LEFT CLICK):

Bits per second:	4800
Data bits:	8
Parity:	None
Stop bits:	1
Flow control:	Xon / Xoff



Bits per second: 4800

Data bits: 8

Parity: None

Stop bits: 1

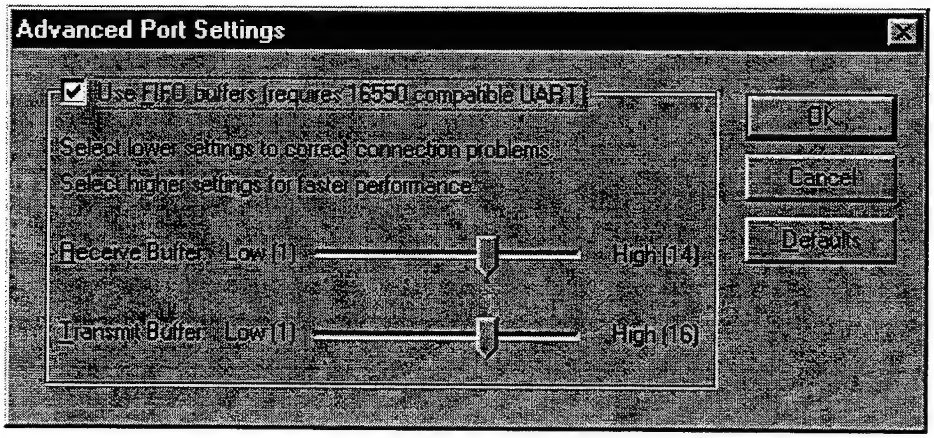
Flow control: Xon / Xoff

- LEFT CLICK the “Advanced” button to open the “Advanced Port Settings” dialog box.

- LEFT CLICK the check box "Use FIFO buffers (requires 16550 compatible UART)". If the box is already has a check mark in it you can ignore this step.

- LEFT CLICK & HOLD DOWN the buton to adjust the slide bars.

- SET the "Receive Buffer:" and the "Transmit Buffer:" to the 3rd tick position from the left as shown by adjusting the slide bars.



☑ Use FIFO buffers (requires 16550 compatible UART)

Select lower settings to correct connection problems.
Select higher settings for faster performance.

Receive Buffer: Low (1) High (14)

Transmit Buffer: Low (1) High (16)

- LEFT CLICK the "OK" button in the "Advanced Port Settings" dialog box. This will close the "Advanced Port Settings" dialog box and return you to the "Communication Port (COM1) Properties" dialog box.

- LEFT CLICK the "OK" button in the "Communication Port (COM1) Properties" dialog box. This will close the "Communication Port (COM1) Properties" dialog box and return you to the "System Properties" dialog box.

- LEFT CLICK the "OK" button in the "System Properties" dialog box. This will close the "System Properties" dialog box and return you to the desktop.

- Now the port is configured to communicate with the KVH Transceiver Unit using standard "AT" modem commands.

2. Using HyperTerminal to test the KVH TU configuration

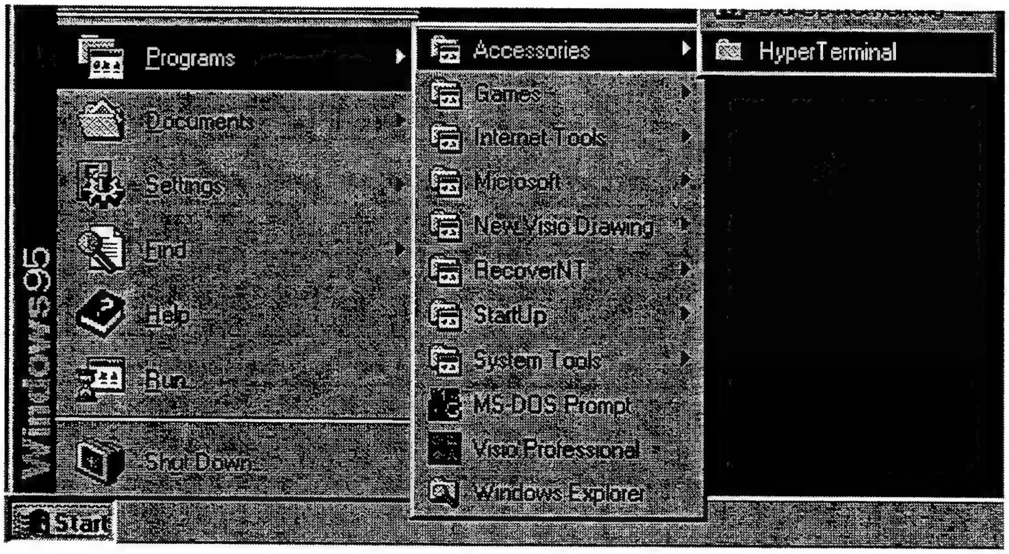
If HyperTerminal is installed on your Win95 computer, then the program will be located in the following location.

C:\Program Files\Accessories\HyperTerminal\Hypertrm.exe.

If HyperTerminal is NOT installed use Windows95 Help to get information on "Adding HyperTerminal".

Running HyperTerminal

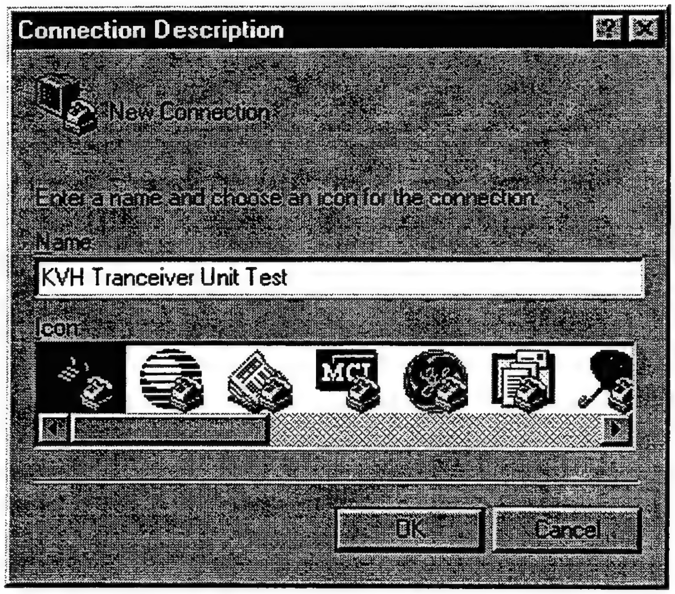
- LEFT CLICK "Start"
- HIGHLIGHT "Programs", HIGHLIGHT "Accessories", HIGHLIGHT "HyperTerminal" and LEFT CLICK to open the "HyperTerminal" folder



- DOUBLE LEFT CLICK on the "Hypertrm.exe" Icon (shown below) in the "HyperTerminal" folder. This will open the "Connection Description" dialog box

Hypertrm.exe

- Enter a name into the "Name" field (Any name). Here the name "KVH Transceiver Unit Test" is used. Pick a pretty icon. LEFT CLICK the "OK" button. This will open the "Phone Number" dialog box.



Connection Description

New Connection

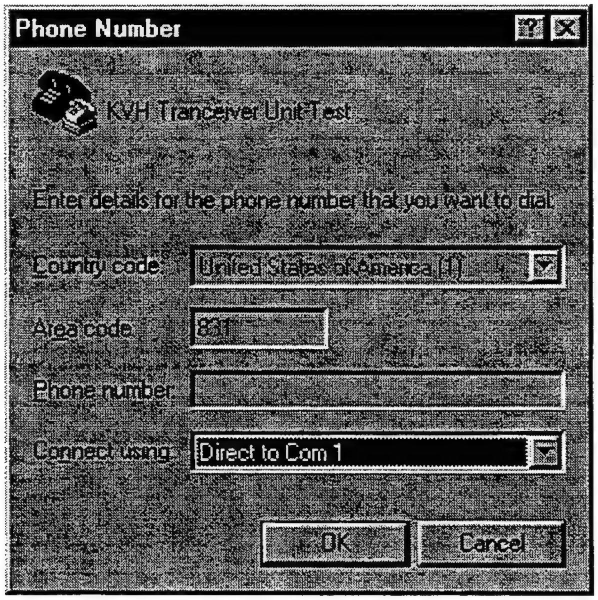
Enter a name and choose an icon for the connection:

Name:

KVH Tranceiver Unit Test

Icon:

- In the “Connect using” drop down box SELECT the COM Port where the KVH TU is connected. Here it is “Direct to Com 1”. Use the drop down box to see all of the possible choices. Once you choose “Direct to Com x” all the other fields will become dark indicating that no other information is needed. LEFT CLICK the OK button. This will open the “COM1 Properties” dialog box.



Phone Number

KVH Tranceiver Unit Test

Enter details for the phone number that you want to dial:

Country code: United States of America (1)

Area code: 831

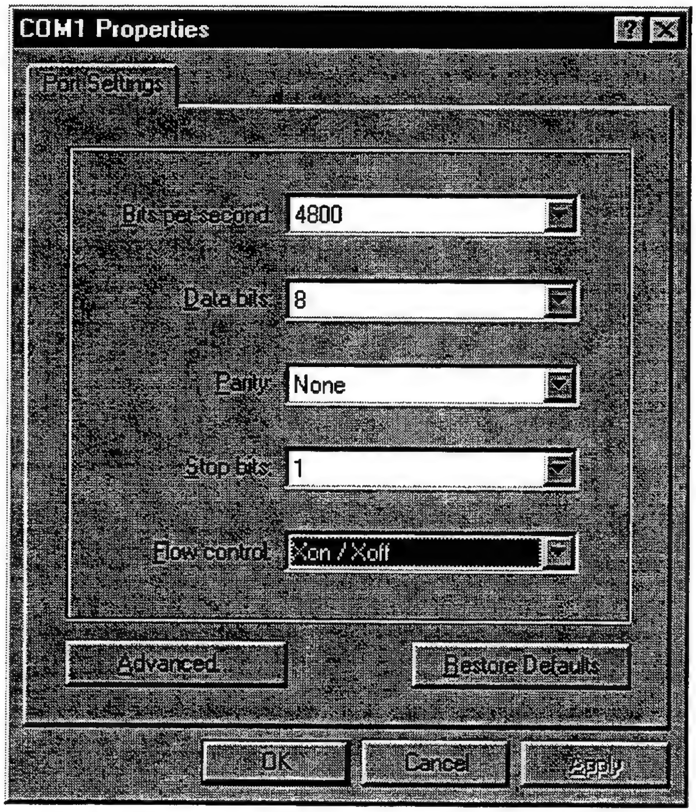
Phone number:

Connect using: Direct to Com 1

- SELECT the following values in the "Port Settings" drop down boxes:

Bits per second:	4800
Data bits:	8
Parity:	None
Stop bits:	1
Flow control:	Xon / Xoff

- LEFT CLICK the "Advanced" button. This will open the "Advanced Port Settings" dialog box.



Bits per second: 4800

Data bits: 8

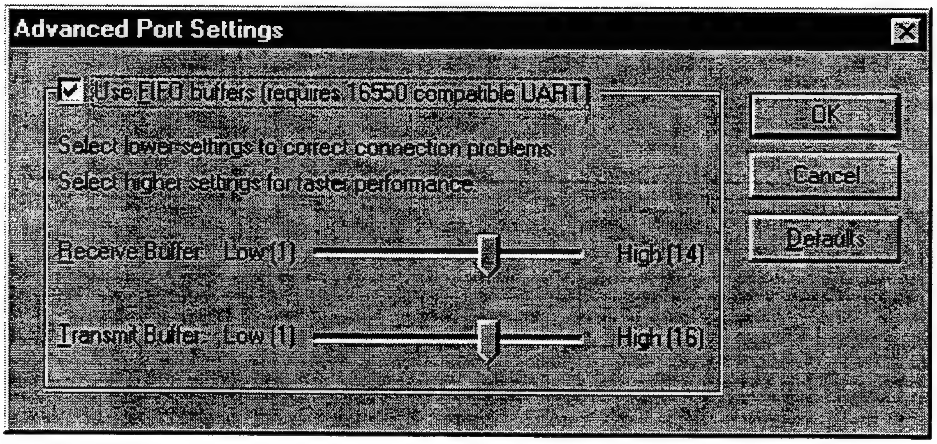
Parity: None

Stop bits: 1

Flow control: Xon / Xoff

- LEFT CLICK the check box "Use FIFO buffers (requires 16550 compatible UART)". If the box is already has a check mark in it you can ignore this step.

- SET the "Receive Buffer:" and the "Transmit Buffer:" to the 3rd tick position from the left as shown below by adjusting the slide bars.



Advanced Port Settings

☑ Use FIFO buffers (requires 16550 compatible UART)

Select lower settings to correct connection problems.

Select higher settings for faster performance.

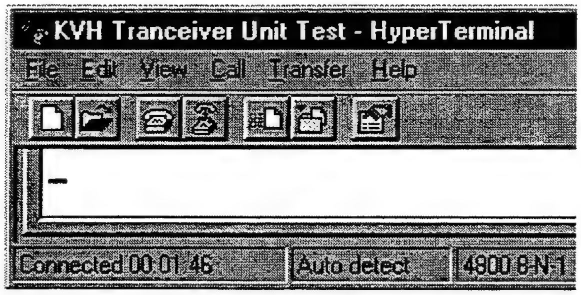
Receive Buffer: Low (1) High (14)

Transmit Buffer: Low (1) High (16)

- LEFT CLICK the "OK" button. This will close the Advanced Port Settings" dialog box and return you to the "COM1 Properties" dialog box.

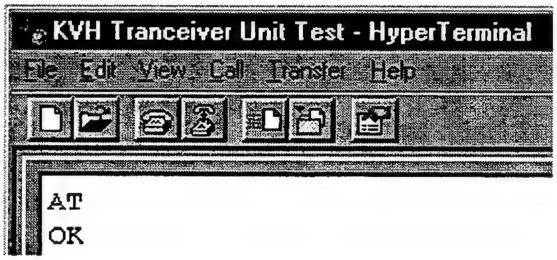
- LEFT CLICK the "OK" button. This will close the "COM1 Properties" dialog box and open the HyperTerminal main screen.

- You are connected now. To verify, look for a timer that is counting in the lower left hand corner. There will also be a blinking cursor.


KVH Tranceiver Unit Test - HyperTerminal
File Edit View Call Transfer Help

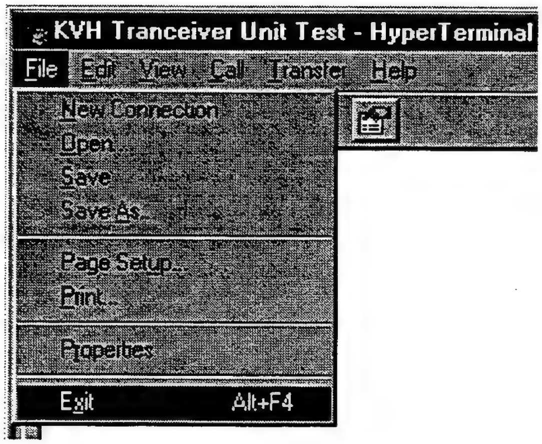
```
Connected 00:01:46    Auto detect    4800 8-N-1
```

- To test your connection TYPE "AT" and hit "ENTER". the screen will respond with "OK". This means that the KVH TU is properly setup for use with your PC. For more commands consult the AMSC Technical Bulletin Subject: "Using Circuit Switched Data on the AMSC System".

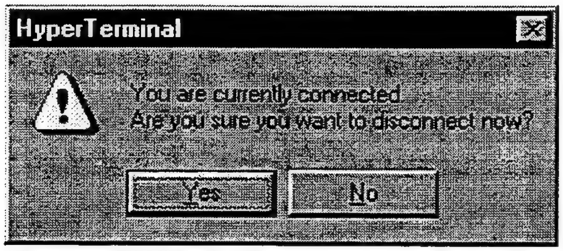

KVH Tranceiver Unit Test - HyperTerminal
File Edit View Call Transfer Help

```
AT
OK
```

- When finished LEFT CLICK "File", HIGHLIGHT "Exit", and LEFT CLICK.

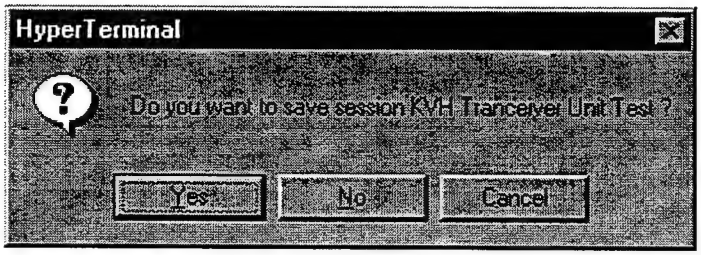


- LEFT CLICK “Yes” to disconnect.



You are currently connected.
Are you sure you want to disconnect now?

- LEFT CLICK “Yes” to save the session settings. this will close Hyperterminal and return you to the desktop.



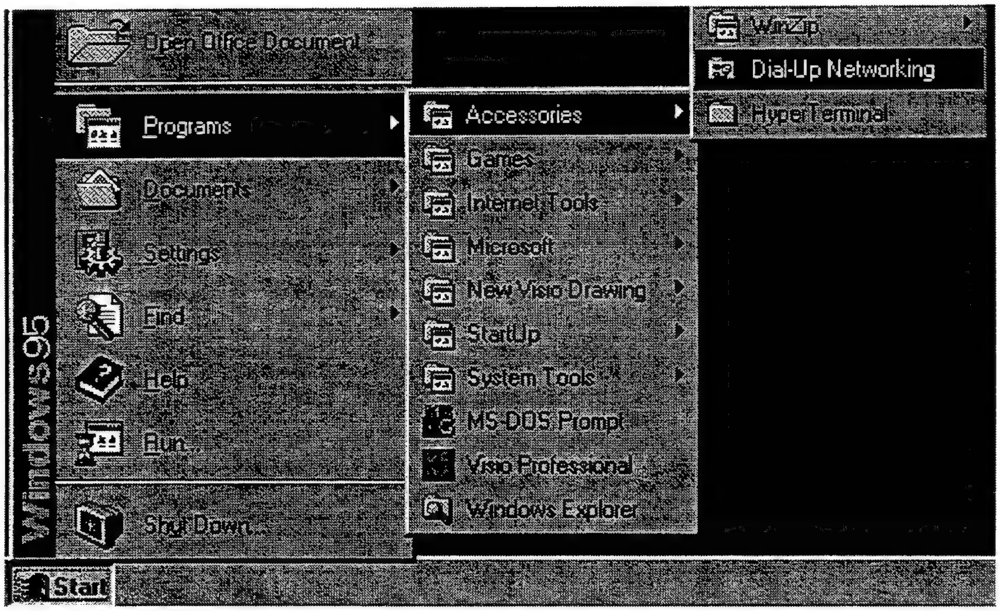
Do you want to save session KVH Tranceiver Unit Test ?

Win95 Dial-Up Networking Setup for a PPP connection with a ISP through the KVH Tracphone

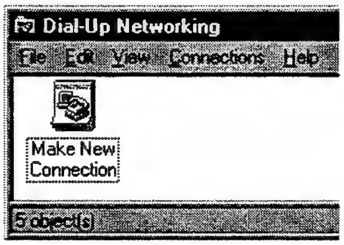
In order to set up Dial-Up Networking you will need the following information:

- A dial up account with an Internet Service Provider (ISP).
- ISP phone number including area code.
- ISP Primary and Secondary Domain Name Server (DNS).
- The login name and password to access the account.

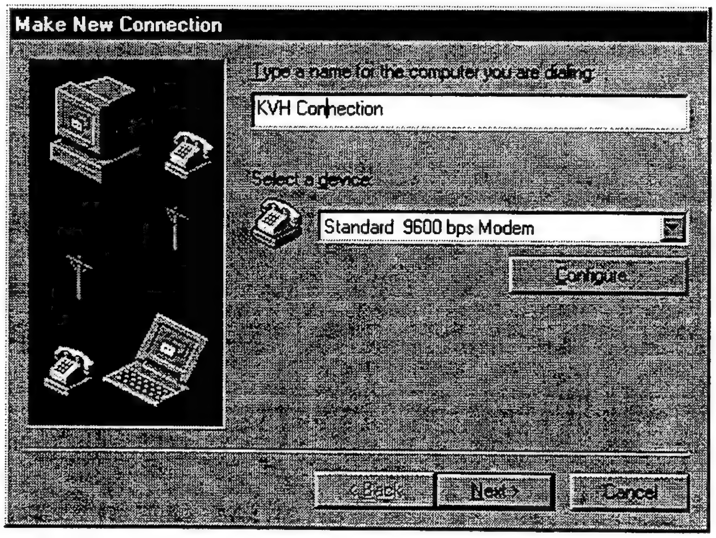
- To start Dial-Up Networking LEFT CLICK "Start", HIGHLIGHT "Accessories", HIGHLIGHT "Dial-Up Networking", and LEFT CLICK. This will open the "Dial-Up Networking" Folder.



- DOUBLE LEFT CLICK on the "Make New Connection" Icon. This will open the "Make New Connection" dialog box.



- TYPE in a name for the connection you are creating. Here it is called "KVH Connection". In the "Select a device" drop down box SELECT "Standard 9600 bps Modem". LEFT CLICK the "Next" button to continue.



Make New Connection

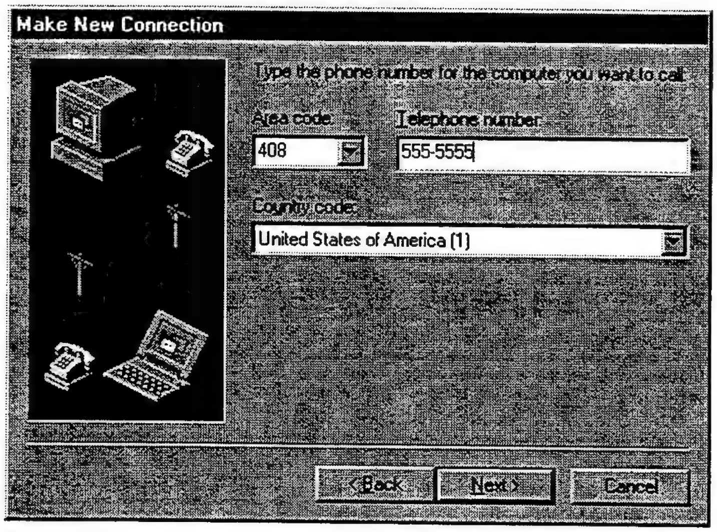
Type a name for the computer you are dialing:

KVH Connection

Select a device:

Standard 9600 bps Modem

- TYPE in the phone number and country code and LEFT CLICK "Next"



Make New Connection

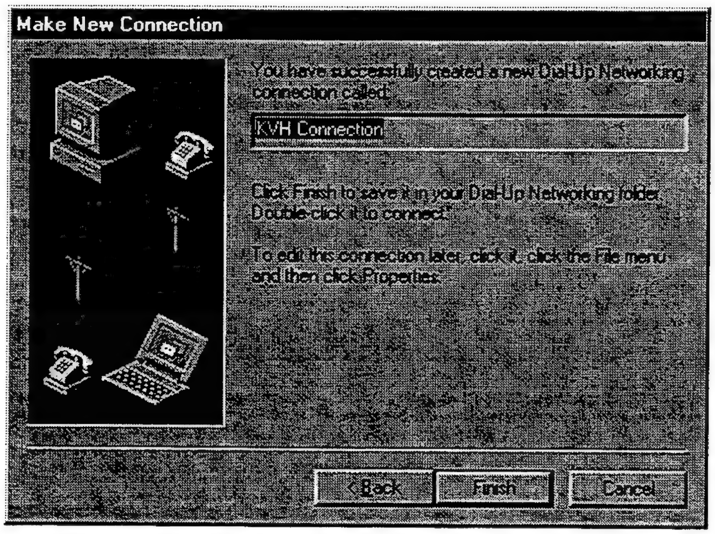
Type the phone number for the computer you want to call:

Area code: 408

Telephone number: 555-5555

Country code: United States of America (1)

- LEFT CLICK "Finish" to complete the initial setup of the Dial-Up Connection. This will return you to the "Dial-Up Networking" folder where the new "KVH Connection" will now be visible.



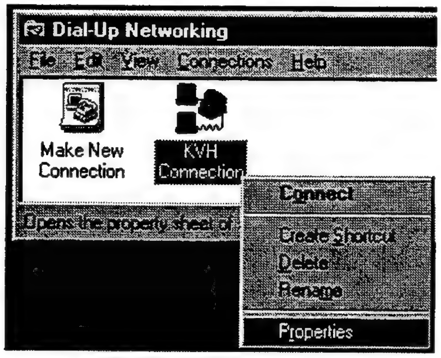
You have successfully created a new Dial-Up Networking connection called:

KVH Connection

Click Finish to save it in your Dial-Up Networking folder. Double-click it to connect!

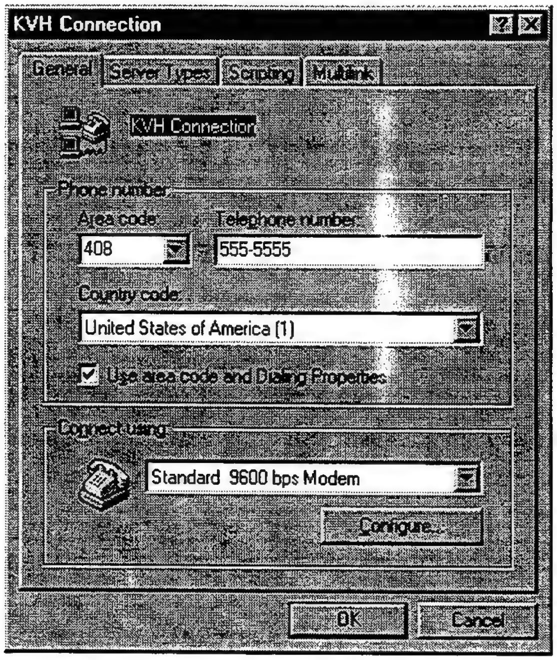
To edit this connection later, click it, click the File menu and then click Properties.

- RIGHT CLICK the new "KVH Connection" icon, HIGHLIGHT "Properties", and LEFT CLICK. This will open the "KVH Connection" dialog box.



Opens the property sheet of

- VERIFY the information and then LEFT CLICK the “Configure” button. this will open the “Standard 9600 modem Properties” dialog box.



KVH Connection

KVH Connection

Phone number

Area code: 408

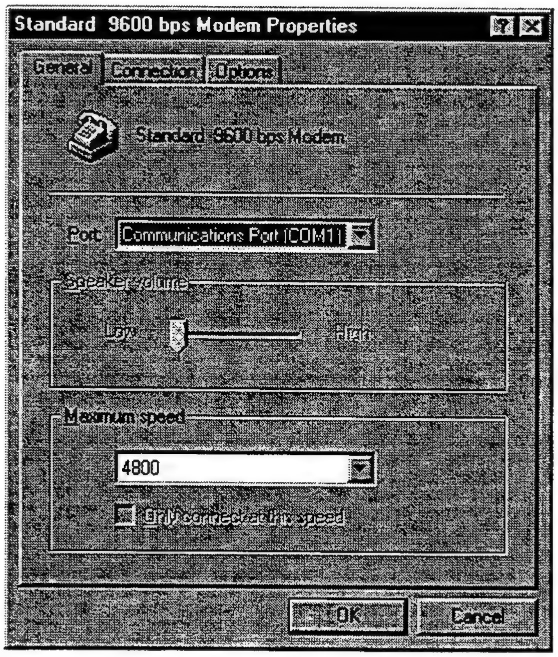
Telephone number: 555-5555

Country code: United States of America (1)

☑ Use area code and Dialing Properties

Connect using: Standard 9600 bps Modem

- VERIFY that the correct COM port is selected in the “Port” field. Here it is “Communication Port (COM1)”. This is the port where the KVH TU is connected. Select the appropriate “Maximum speed”, for the KVH Tracphone is 4800 bps, so select 4800. LEFT CLICK the “Connection” tab.



Standard 9600 bps Modem Properties

Standard 9600 bps Modem

Port: Communications Port (COM1)

Speaker volume: Low — High

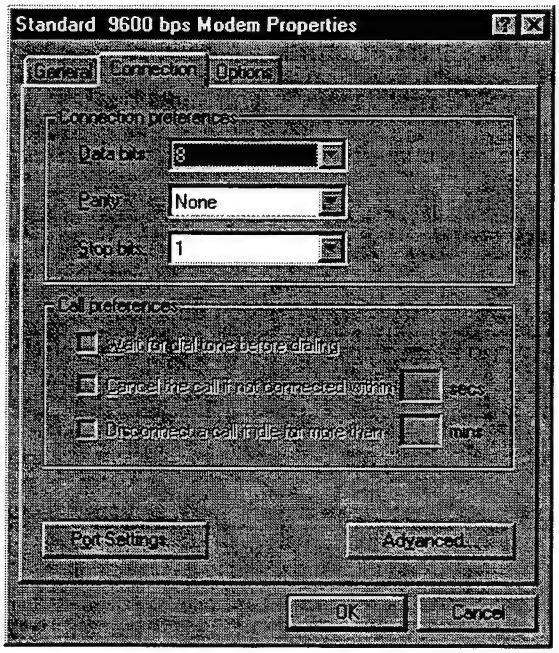
Maximum speed: 4800

□ Only connect at this speed

- SELECT the following values in the "Connection" tab drop down boxes:

Data bits:	8
Parity:	None
Stop bits:	1

- LEFT CLICK the "Port Settings' button. This will open the "Advanced Port Settings" dialog box.



Standard 9600 bps Modem Properties

Connection preferences

Data bits: 8

Parity: None

Stop bits: 1

Call preferences

□ Wait for dial tone before dialing

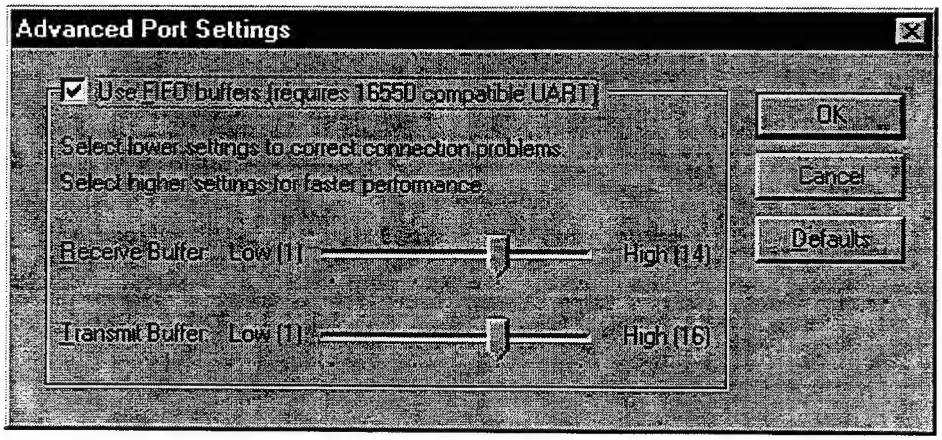
□ Cancel the call if not connected within □ secs

□ Disconnect a call if idle for more than □ mins

- LEFT CLICK the check box "Use FIFO buffers (requires 16550 compatible UART)". If the box is already has a check mark in it you can ignore this step.

- SET the "Receive Buffer:" and the "Transmit Buffer:" to the 3rd tick position from the left as shown below by adjusting the slide bars.

- LEFT CLICK the "OK " button. This will return you to the "Standard 9600 bps Modem Properties" dialog box and the "Connection" tab.



Advanced Port Settings

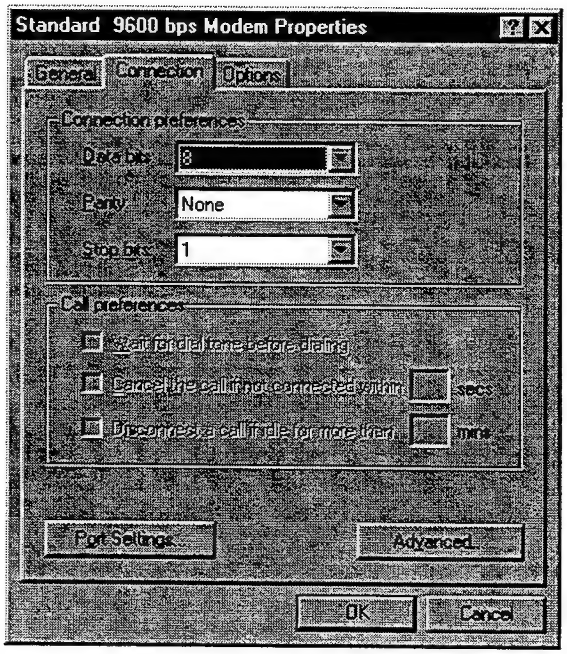
☑ Use FIFO buffers (requires 16550 compatible UART)

Select lower settings to correct connection problems.
Select higher settings for faster performance.

Receive Buffer: Low (1) High (14)

Transmit Buffer: Low (1) High (16)

- LEFT CLICK the "Advanced" button. This will open the "Advanced Connection Settings" dialog box.



Standard 9600 bps Modem Properties

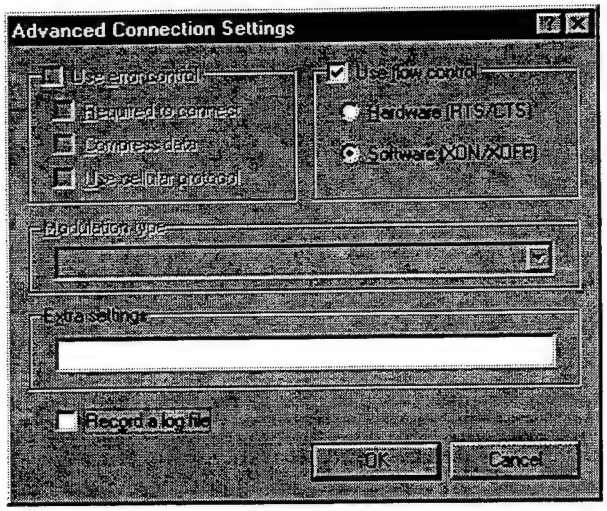
Connection preferences

Data bits: 8
Parity: None
Stop bits: 1

Call preferences

□ Wait for dial tone before dialing
□ Cancel the call if not connected within secs
□ Disconnect a call if idle for more than mins

- LEFT CLICK "Use Flow Control" check box and LEFT CLICK the "Software (XON/XOFF)"option button. LEFT CLICK the "OK" button. This will return you to the "Standard 9600 bps Modem Properties" dialog box and the "Connection" tab.



Advanced Connection Settings

□ Use error control

□ Required to connect

□ Compress data

□ Use cellular protocol

☑ Use flow control

○ Hardware (RTS/CTS)

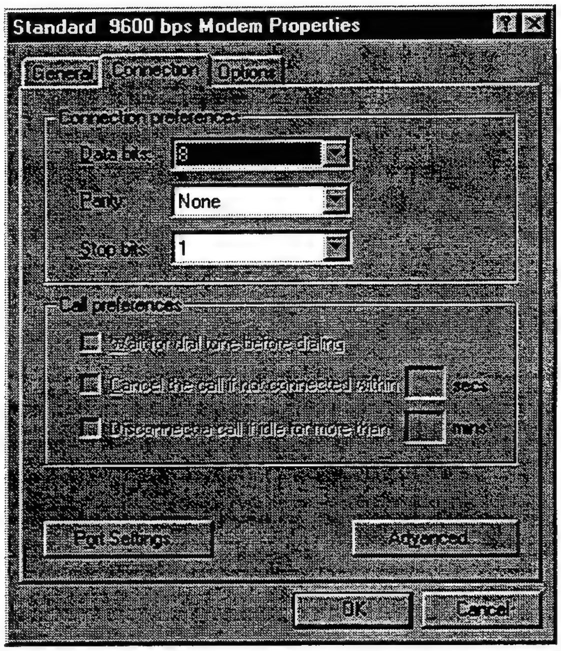
● Software (XON/XOFF)

Modulation type

Extra settings

□ Record a log file

- LEFT CLICK the “Options” tab



Standard 9600 bps Modem Properties

Connection preferences

Data bits: 8

Parity: None

Stop bits: 1

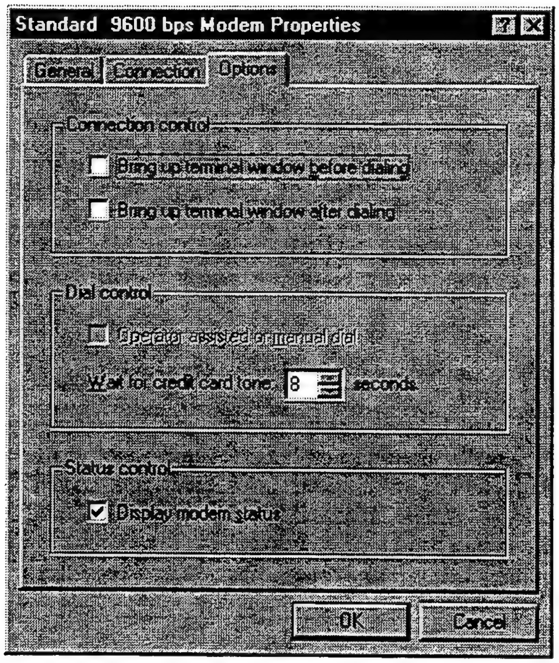
Call preferences

□ Wait for dial tone before dialing

□ Cancel the call if not connected within secs

□ Disconnect a call if idle for more than mins

- LEFT CLICK the “Display modem status” checkbox. NOTE! If you are connecting to a UNIX based system LEFT CLICK the “Bring up terminal window after dialing” checkbox, otherwise leave it blank. LEFT CLICK the “OK” button. This will return you to the “KVH Connection” dialog box.



Standard 9600 bps Modem Properties

Connection control

□ Bring up terminal window before dialing

□ Bring up terminal window after dialing

Dial control

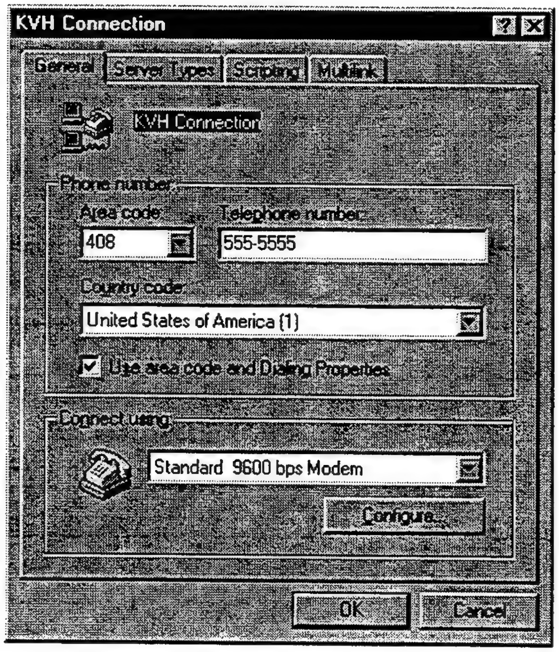
□ Operator assisted or manual dial

Wait for credit card tone: 8 seconds

Status control

☑ Display modem status

- LEFT CLICK the "Server Types" tab.



KVH Connection

KVH Connection

Phone number:

Area code: 408

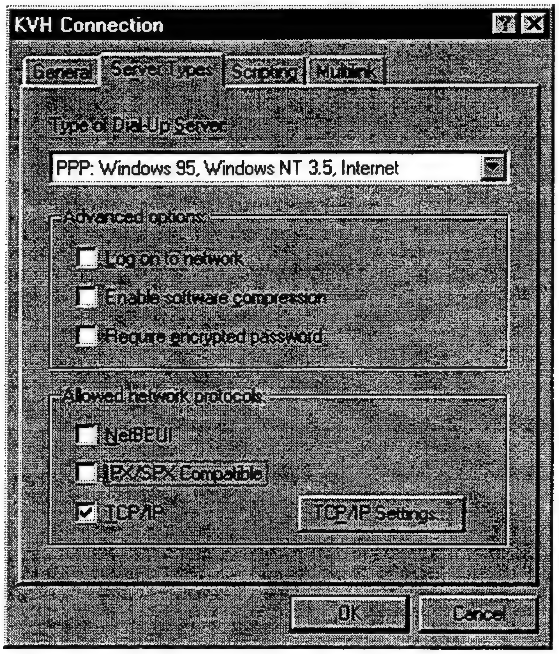
Telephone number: 555-5555

Country code: United States of America (1)

☑ Use area code and Dialing Properties

Connect using: Standard 9600 bps Modem

- SELECT "PPP: Windows 95, Windows NT 3.5 Internet" in the "Type of Dial-Up Server:" drop down box. LEFT CLICK the TCP/IP checkbox under the "Allowed network protocols" section. LEFT CLICK the "TCP/P Settings button. This will open the "TCP/IP Settings" dialog box.



KVH Connection

General | Server Types | Scripting | Multilink

Type of Dial-Up Server:

PPP: Windows 95, Windows NT 3.5, Internet

Advanced options:

□ Log on to network

□ Enable software compression

□ Require encrypted password

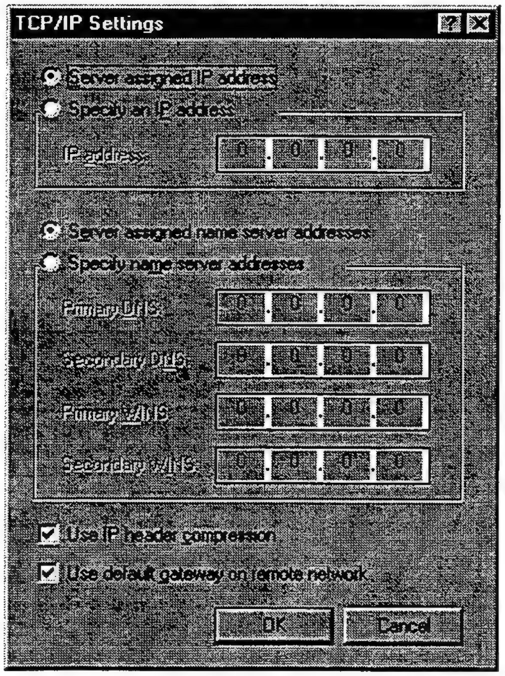
Allowed network protocols:

□ NetBEUI

□ IPX/SPX Compatible

☑ TCP/IP

- LEFT CLICK the “Server assigned IP address” option button. LEFT CLICK the “Server assigned name server address” option button. LEFT CLICK the “Use IP header compression” check box. LEFT CLICK the “Use default gateway on remote network” check box. LEFT CLICK the “OK” button. This will return you to the “KVH Connection” dialog box.



TCP/IP Settings

● Server assigned IP address

○ Specify an IP address

IP address: 0 . 0 . 0 . 0

● Server assigned name server addresses

○ Specify name server addresses

Primary DNS: 0 . 0 . 0 . 0

Secondary DNS: 0 . 0 . 0 . 0

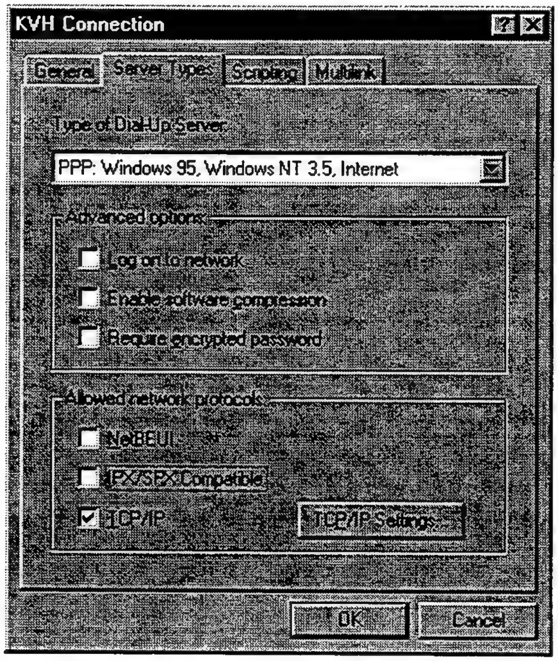
Primary WINS: 0 . 0 . 0 . 0

Secondary WINS: 0 . 0 . 0 . 0

☑ Use IP header compression

☑ Use default gateway on remote network

- LEFT CLICK the "OK" button. This will return you to the "Dial-Up Networking" folder

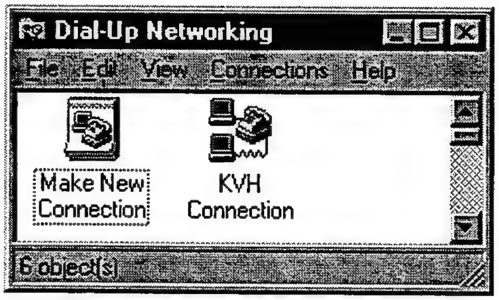


Type of Dial-Up Server:
PPP: Windows 95, Windows NT 3.5, Internet

Advanced options:
□ Log on to network
□ Enable software compression
□ Require encrypted password

Allowed network protocols:
□ NetBEUI
□ IPX/SPX Compatible
☑ TCP/IP

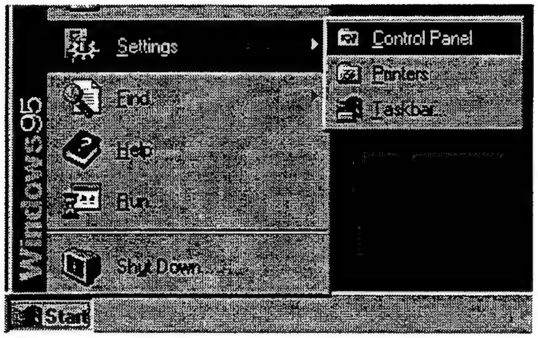
- Dial-Up Networking for the KVH Tracphone is complete. To connect DOUBLE LEFT CLICK on the "KVH Connection" icon within the "Dial-Up Networking" folder.



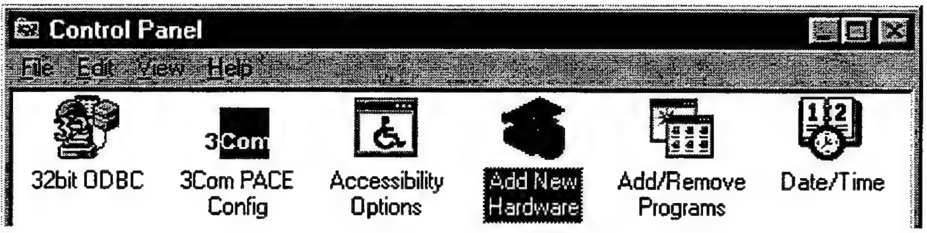
6 object(s)

Adding a Standard modem to Win95 to utilize Dial-Up Networking

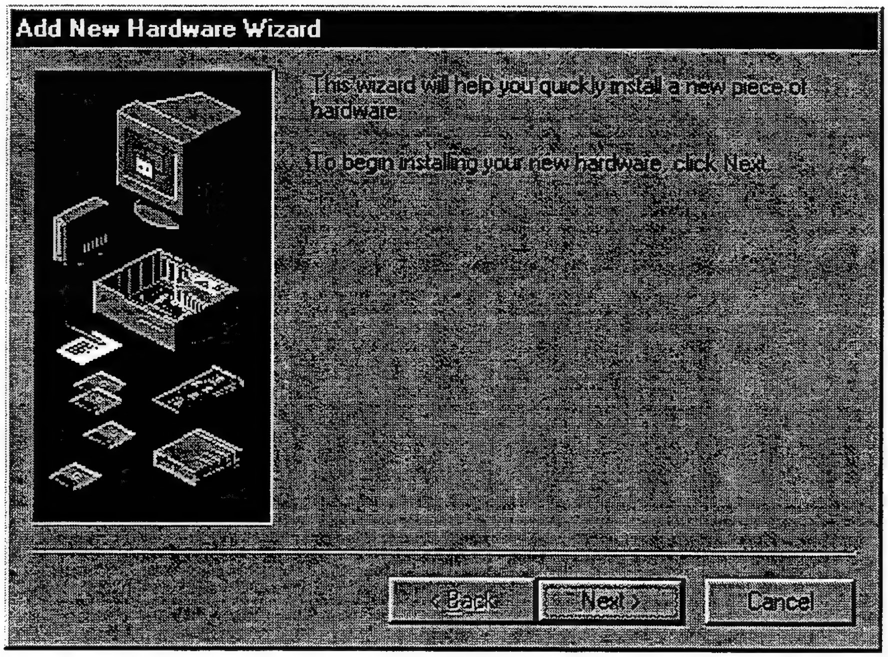
- LEFT CLICK "Start", HIGHLIGHT "Control Panel", and LEFT CLICK. This will open the "Control Panel" folder.



- DOUBLE LEFT CLICK the "Add New Hardware" icon. This will open the "Add New Hardware Wizard" dialog box.



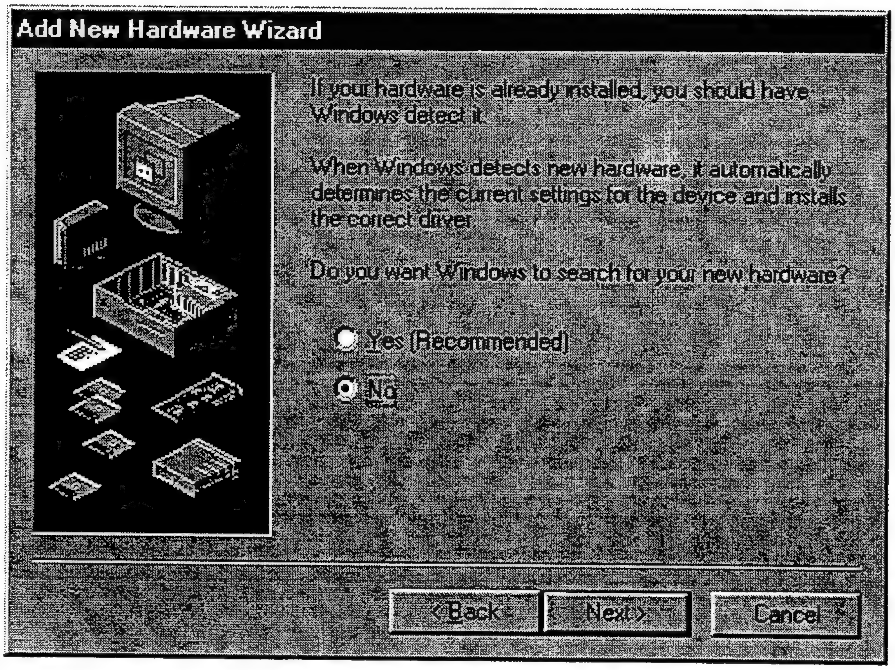
- LEFT CLICK the "Next" button to continue.



This wizard will help you quickly install a new piece of hardware.

To begin installing your new hardware, click Next.

- LEFT CLICK the “No” check box so Win95 will not search for new hardware. LEFT CLICK the “Next” button to continue.



Add New Hardware Wizard

If your hardware is already installed, you should have Windows detect it.

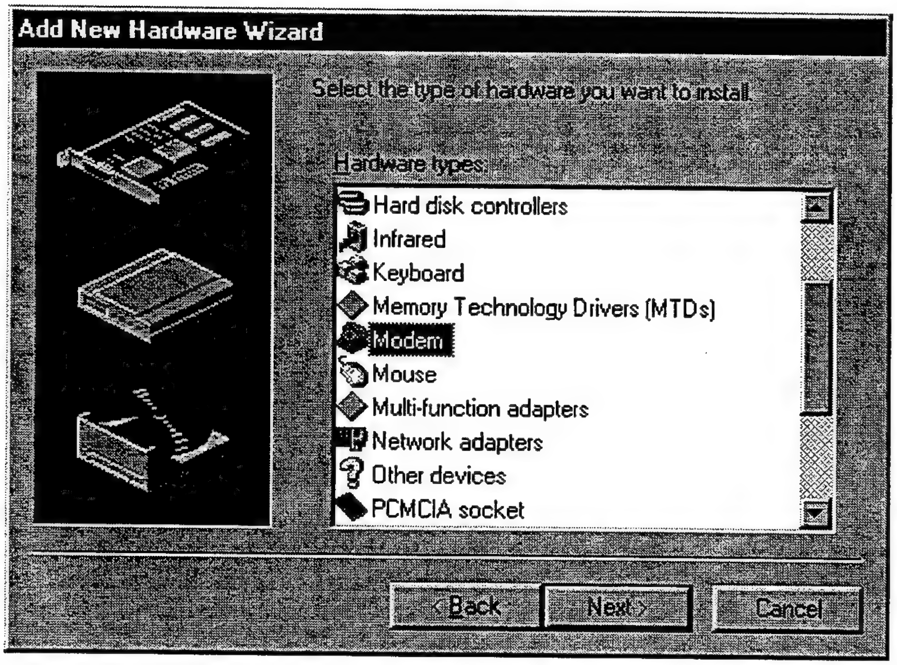
When Windows detects new hardware, it automatically determines the current settings for the device and installs the correct driver.

Do you want Windows to search for your new hardware?

○ Yes (Recommended)

● No

- LEFT CLICK the “Modem” icon to HIGHLIGHT it. LEFT CLICK the “Next” button to continue.



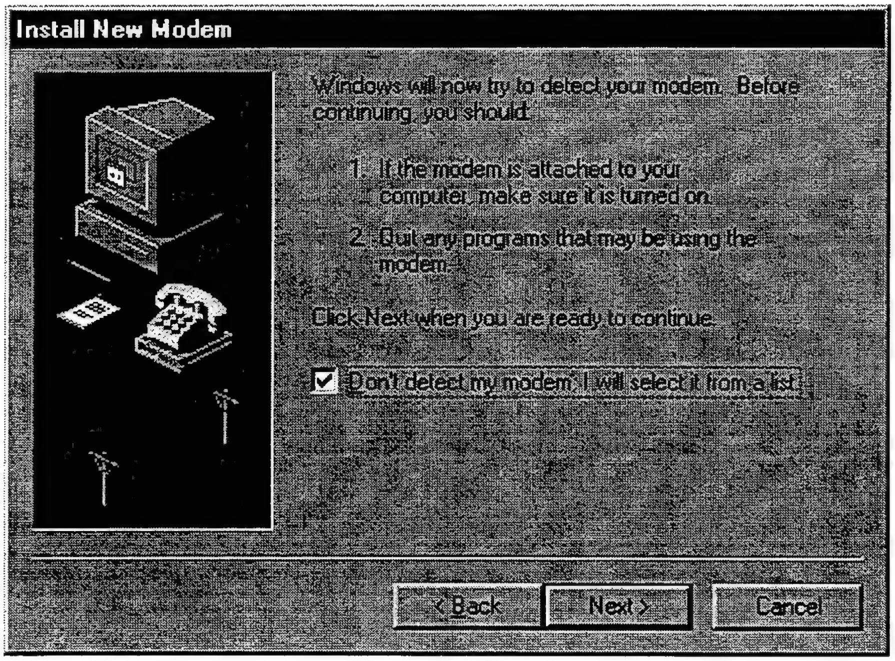
Add New Hardware Wizard

Select the type of hardware you want to install.

Hardware types:

- Hard disk controllers
- Infrared
- Keyboard
- Memory Technology Drivers (MTDs)
- Modem
- Mouse
- Multi-function adapters
- Network adapters
- Other devices
- PCMCIA socket

- LEFT CLICK the “Don’t detect my modem; I will select it from a list.” check box. LEFT CLICK the “Next” button to continue.



Install New Modem

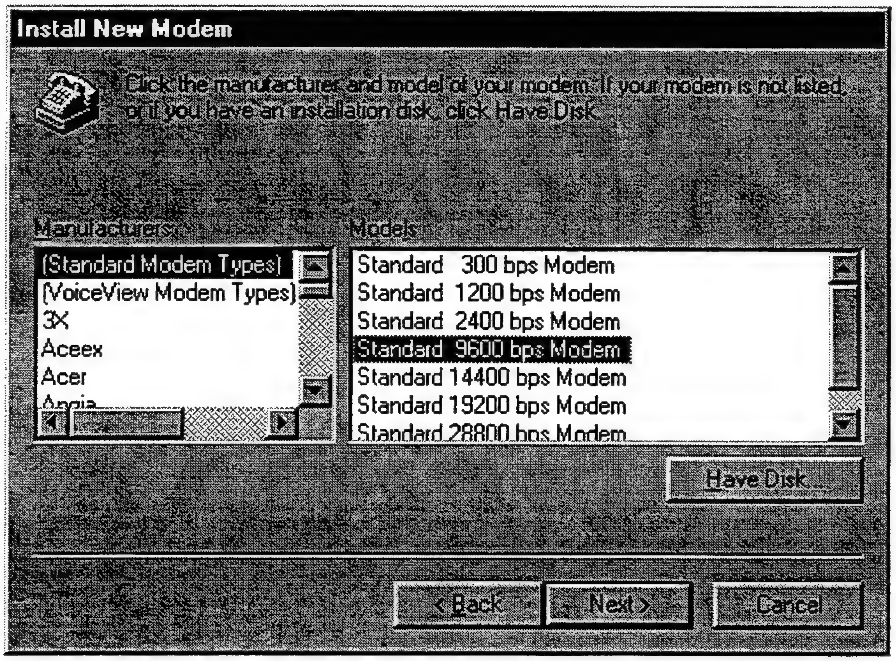
Windows will now try to detect your modem. Before continuing, you should:

1. If the modem is attached to your computer, make sure it is turned on.
2. Quit any programs that may be using the modem.

Click Next when you are ready to continue.

☑ Don't detect my modem; I will select it from a list.

- LEFT CLICK the “(Standard Modem Types)” in the “Manufacturers” (Left) column to HIGHLIGH it. LEFT CLICK the “Standard 9600 bps Modem” in the “Models” column (Right) to HIGHLIGHT it. LEFT CLICK the “Next” button to continue.



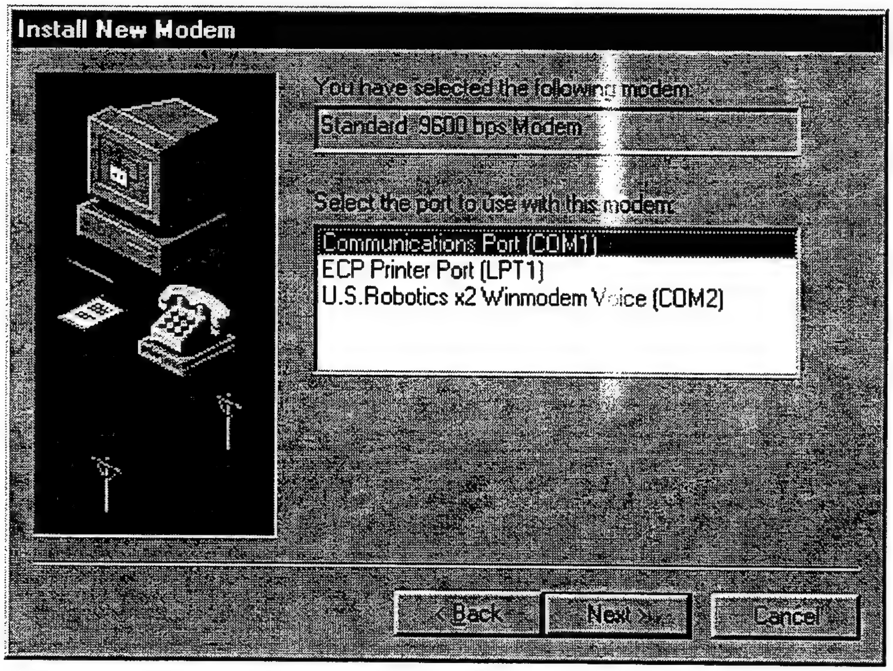
Install New Modem

Click the manufacturer and model of your modem. If your modem is not listed, or if you have an installation disk, click Have Disk.

Manufacturers:
(Standard Modem Types)
(VoiceView Modem Types)
3X
Aceex
Acer
Angia

Models
Standard 300 bps Modem
Standard 1200 bps Modem
Standard 2400 bps Modem
Standard 9600 bps Modem
Standard 14400 bps Modem
Standard 19200 bps Modem
Standard 28800 bps Modem

- LEFT CLICK "Communications Port (COM1)" in the "Select the port to use with this modem" to HIGHLIGHT it. (NOTE! Select the port on your computer where the cable to the KVH Tracphone is connected.) LEFT CLICK the "Next" button to continue.



Install New Modem

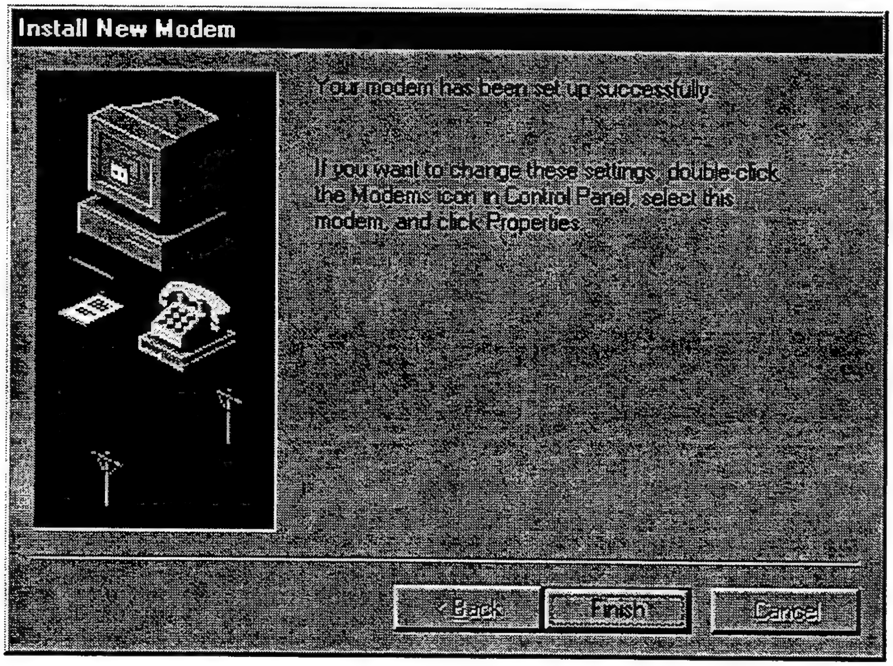
You have selected the following modem:

Standard 9600 bps Modem

Select the port to use with this modem:

Communications Port (COM1)
ECP Printer Port (LPT1)
U.S.Robotics x2 Winmodem Voice (COM2)

- LEFT CLICK the "Finish" button to continue.



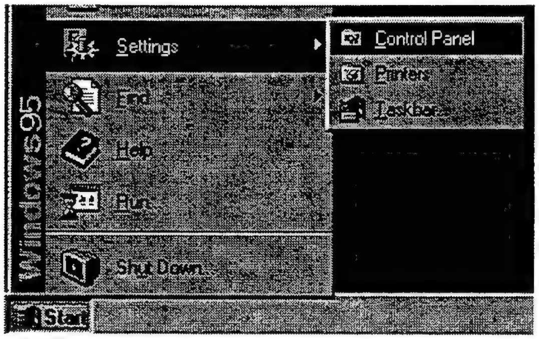
Install New Modem

Your modem has been set up successfully.

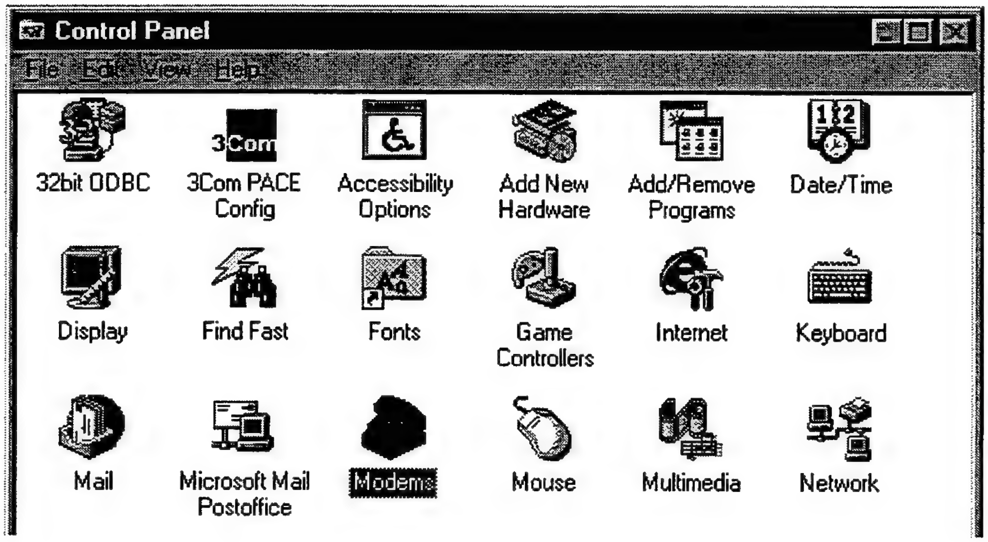
If you want to change these settings, double-click the Modems icon in Control Panel, select this modem, and click Properties.

Verifying the Modem Setup in Control Panel

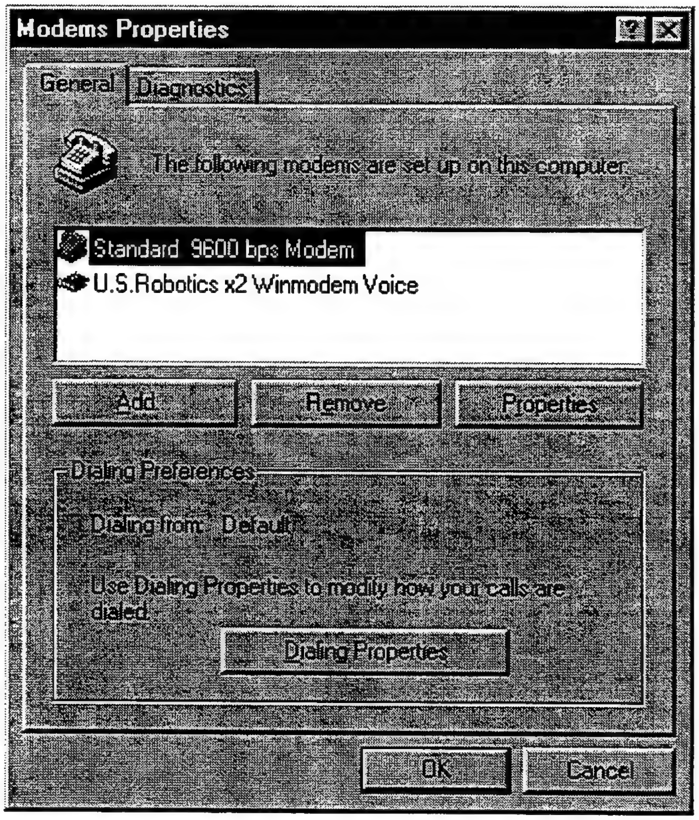
- LEFT CLICK "Start", HIGHLIGHT "Control Panel", and LEFT CLICK. This will open the "Control Panel" folder.



- DOUBLE LEFT CLICK the "Modems" icon to open the "Modem Properties" dialog box.



- Verify that the "Standard 9600 bps Modem" is listed under the "The following modems are set up on this computer." LEFT CLICK the "OK" button to return to the "Control Panel" folder.



The following modems are set up on this computer.

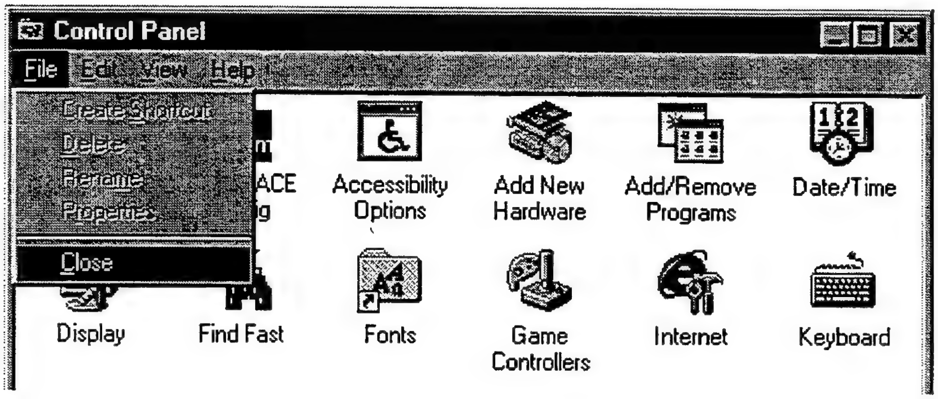
Standard 9600 bps Modem
U.S.Robotics x2 Winmodem Voice

Dialing Preferences

Dialing from: Default

Use Dialing Properties to modify how your calls are dialed.

- LEFT CLICK "File" from the menu, HIGHLIGHT "Close", and then LEFT CLICK to close the "Control Panel". This will return you to the desktop.



APPENDIX B. REDHAT LINUX 5.0 CONFIGURATION

Linux is one of several Operating Systems supported by SeaNet. The SCN software, however, is designed specifically for Linux (particularly the Red Hat version). Linux is a freely distributed PC-based UNIX clone developed by Linus Torvalds in 1991 at the University of Helsinki, Finland [Ref. 19]. Linux is a complete multitasking; multi-user operating system that was designed to maximize the capabilities of Intel based computer architecture[Ref. 19]. Some advantages of the Linux operating system are as follows:

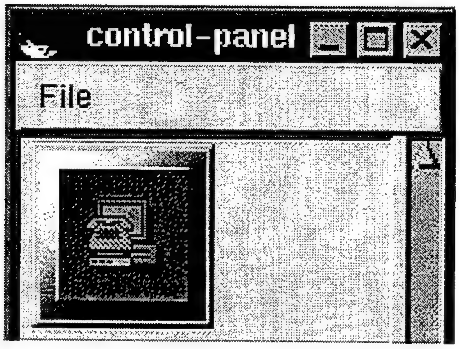
- Freely distributed.
- UNIX features.
- 32 bit, Multi-user, and Multi-tasking.
- Well documented.
- Compatible with Windows, MS-DOS, UNIX.
- Compact.

This remainder of this appendix will discuss how to:

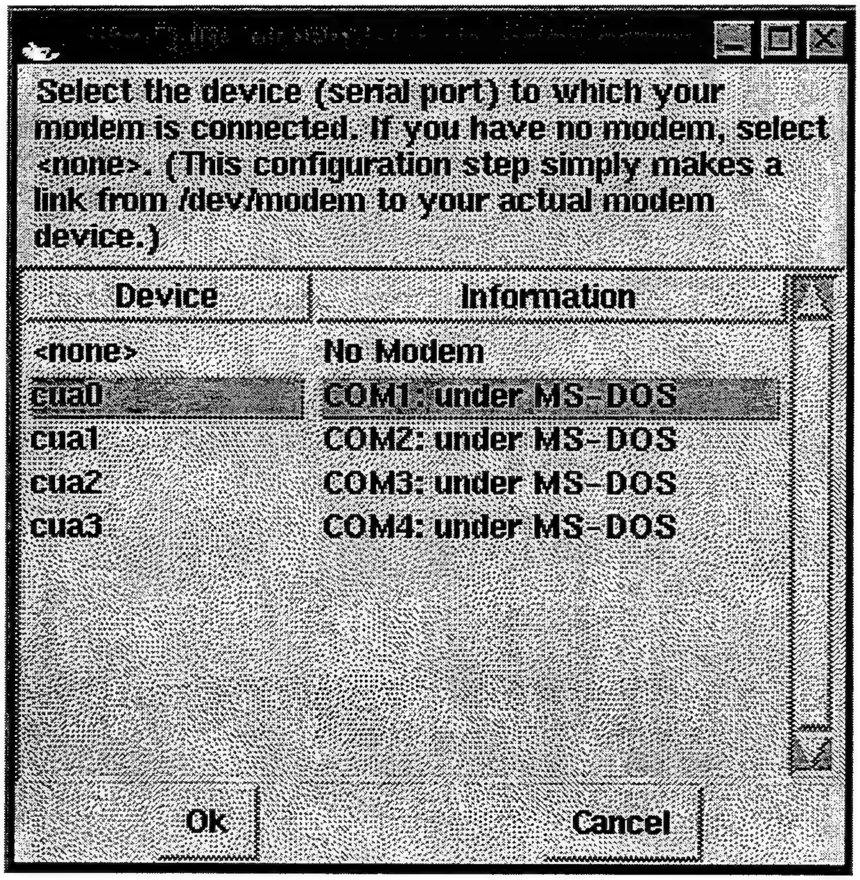
1. Configure a Point-To-Point (PPP) interface using Red Hat Linux 5.0 to allow Point-to-Point connections with an ISP.
2. Configure Red Hat Linux 5.0 to access the Internet through a PPP interface.
3. Turn specific interfaces on/off using the "Usenet" program.
4. Retrieve a dynamically assigned IP from an ISP using the nestat command.
5. Start an Apache HTTP server.
6. Running the SeaNet Communication Node (SCN) software

1. Configuring a point-to-point (PPP) interface.

- LEFT CLICK the Modem button (shown below) from the “Control-panel”. This will open the “Configure Modem” dialog box.



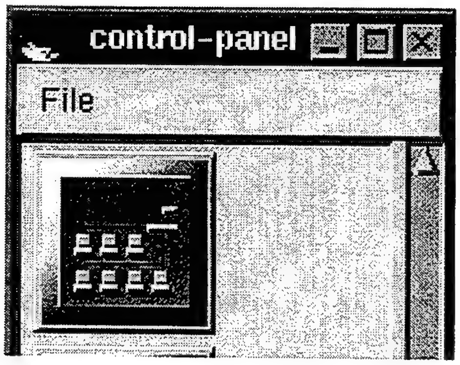
- LEFT CLICK on the "Device" that corresponds to the COM port where the KVH TU is connected. This will HIGHLIGHT the "Device" and its "Information".
- LEFT CLICK the "OK" button when finished. This will close the "Configure Modem" dialog box.



Select the device (serial port) to which your modem is connected. If you have no modem, select <none>. (This configuration step simply makes a link from /dev/modem to your actual modem device.)

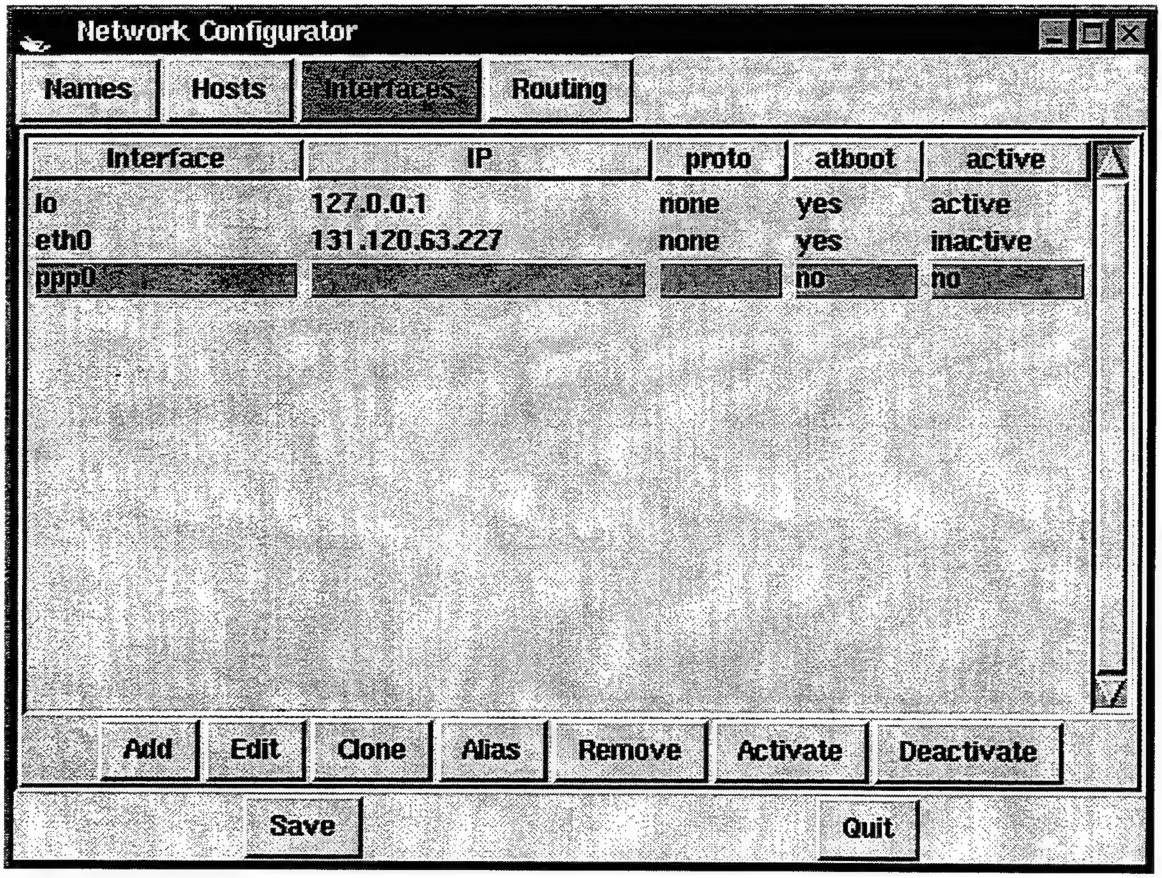
Device	Information
<none>	No Modem
cua0	COM1: under MS-DOS
cua1	COM2: under MS-DOS
cua2	COM3: under MS-DOS
cua3	COM4: under MS-DOS

- LEFT CLICK the Network" button (shown below) icon from the "Control-panel". This will open the "Network Configurator-Names" dialog box.



- LEFT CLICK the “Interfaces” button to open the “Network Configurator-Interfaces” dialog box.

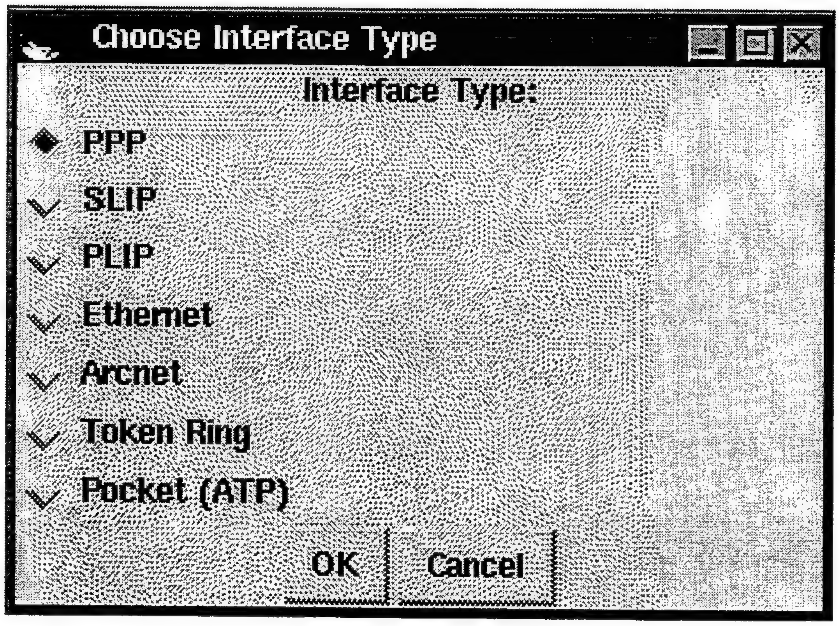
- This dialog box displays the interfaces that exist on the system. For the example below there is a default loopback interface (lo) that cannot be altered. There is 1 Ethernet interface (eth0), and 1 PPP interface (ppp0).



Interface	IP	proto	atboot	active
lo	127.0.0.1	none	yes	active
eth0	131.120.63.227	none	yes	inactive
ppp0			no	no

- LEFT CLICK the “Add” Button to open the “Choose Interface Type” dialog box.

- LEFT CLICK the “PPP” (Point-To-Point) check box.

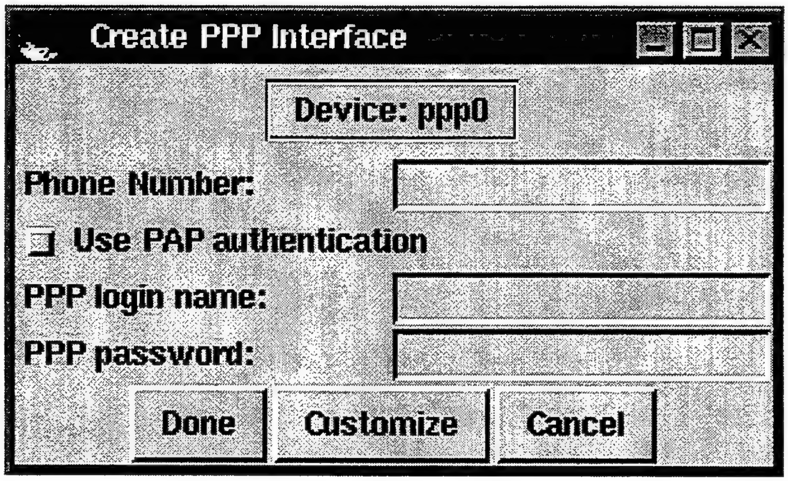


- LEFT CLICK the “OK” button. This will open the “Create PPP Interface” dialog box.

- The "Device: ppp2" represents the third PPP interface for the example below. The interfaces are labeled as "pppX" where "X" is the number of the interface beginning at 0. The first interface would be "ppp0", the second "ppp1", the third "ppp2", etc.

- Enter the information specific to your connection i.e., Phone Number, login name, and password. If your ISP uses PAP (Password Authentication Protocol).

- LEFT CLICK the "Use PAP authentication" check box. If your not sure leave it blank and ask your ISP, you can change the setting later if needed.



Create PPP Interface

Device: ppp0

Phone Number:

□ Use PAP authentication

PPP login name:

PPP password:

- LEFT CLICK the "Customize" button if further customization is required (i.e., UNIX shell login). This will open the "Edit PPP Interface-Hardware" dialog box.

- Match the settings as shown below.

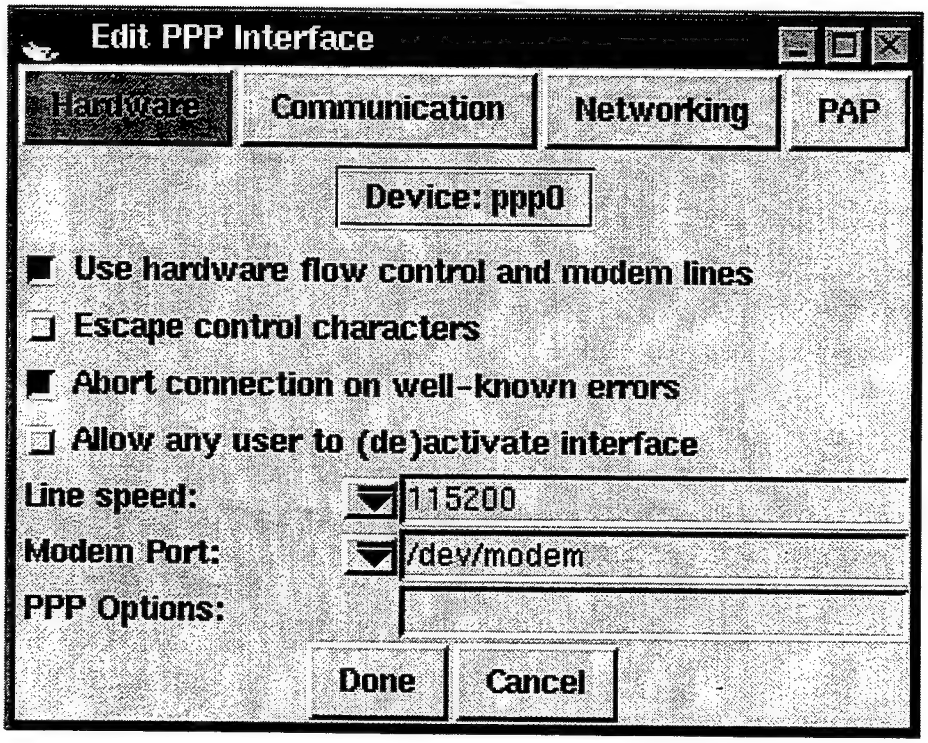
- LEFT CLICK the “Use Hardware flow control and modem lines” check box.

- LEFT CLICK the “Abort connection on well known errors” check box.

- LEFT CLICK the “Allow any user to (de)activate interface” check box.

- LEFT CLICK the drop down box arrow for “Line speed:” and SELECT 4800 from the choices

- VERIFY that the “Modem Port:” field is as shown below “/dev/modem”. This is the default setting and it should not be altered.



Edit PPP Interface

Device: ppp0

☒ Use hardware flow control and modem lines

☐ Escape control characters

☒ Abort connection on well-known errors

☐ Allow any user to (de)activate interface

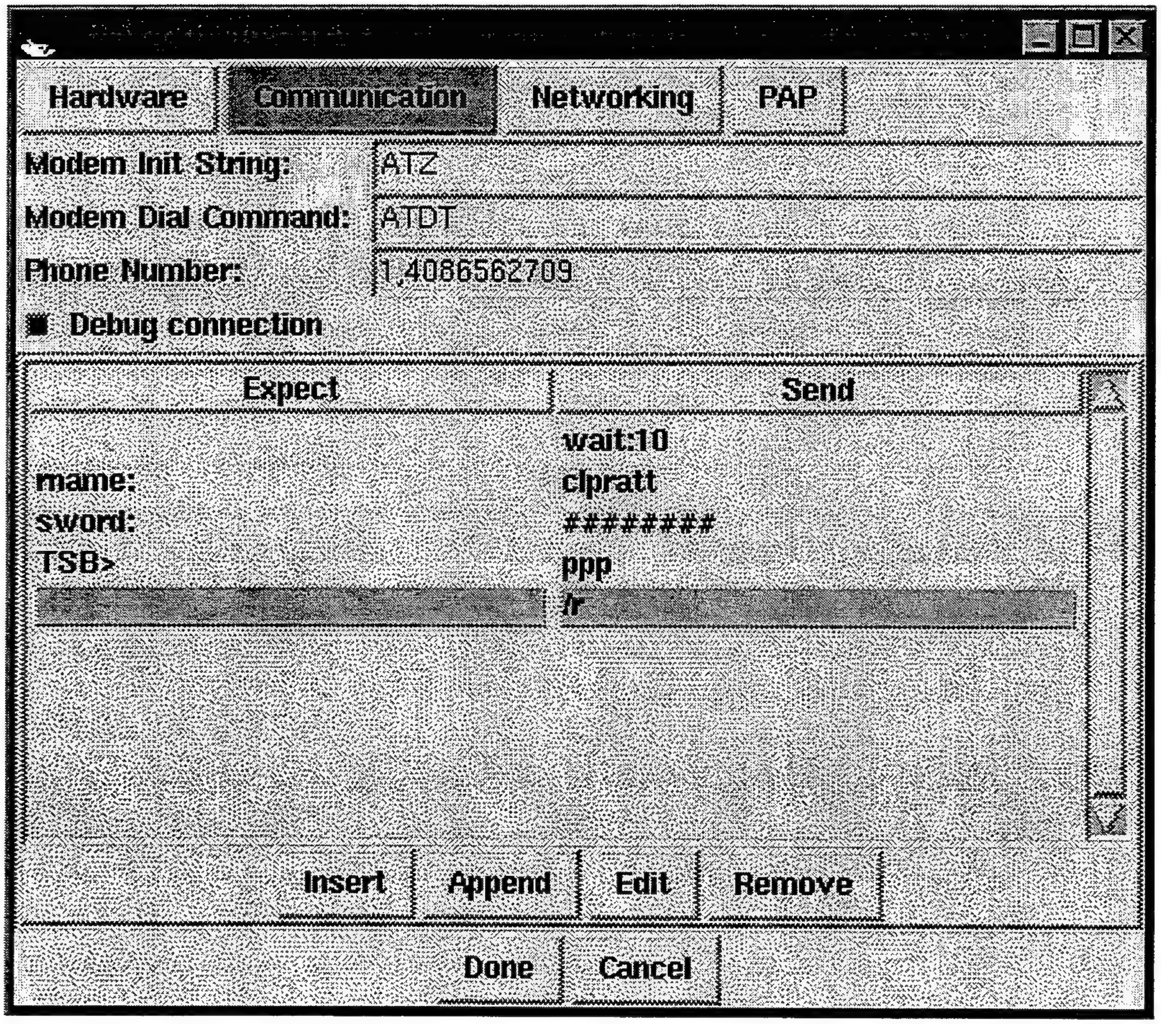
Line speed: 115200

Modem Port: /dev/modem

PPP Options:

- LEFT CLICK the “Communication” button when finished. This will open the “Edit PPP Interface–Communication” dialog box

- The “Edit PPP Interface-Communication” dialog box allows you to add chat scripting to customize a connection. This is useful when logging into a Unix shell where more information than just name and password are needed. Enter the information specific to your connection.



Modem Init String: ATZ

Modem Dial Command: ATDT

Phone Number: 1,4086562709

■ Debug connection

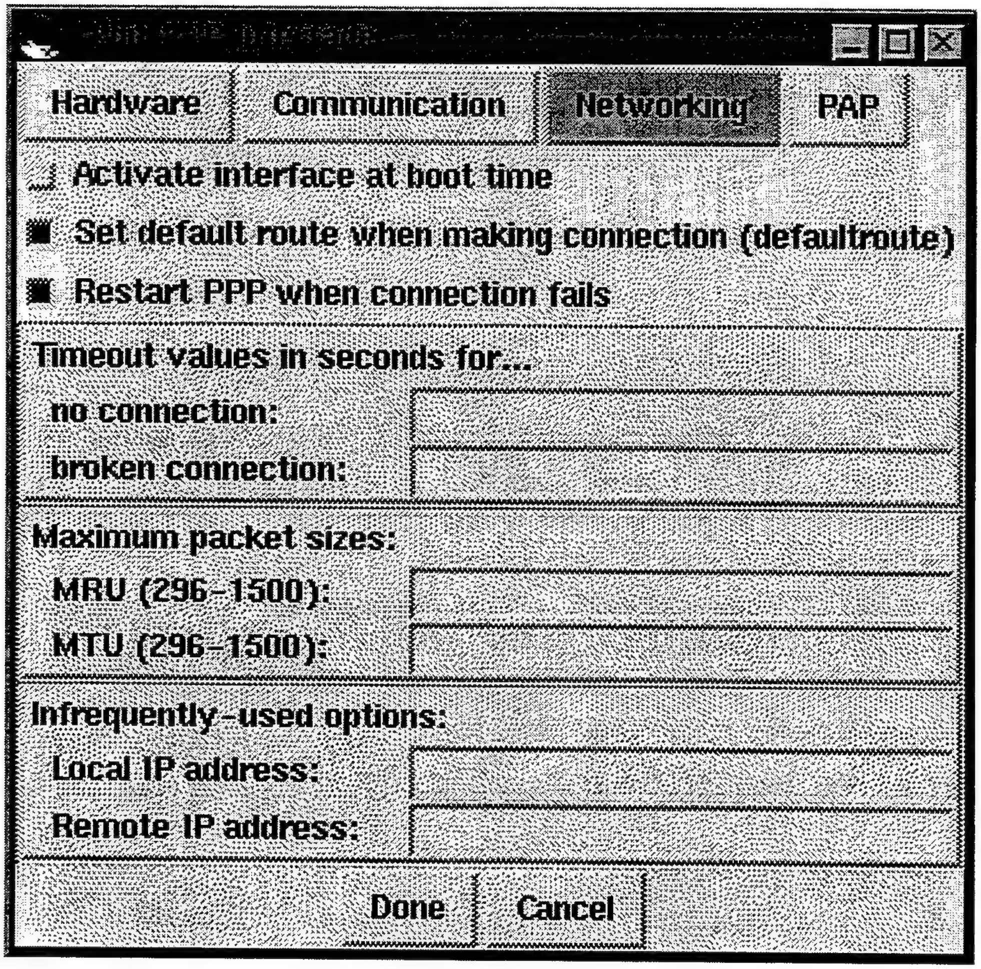
Expect	Send
	wait:10
mame:	clpratt
sword:	########
TSB>	ppp
	/r

- LEFT CLICK the “Networking” button to open the “Edit PPP Interface-Networking“ dialog box.

- LEFT CLICK the “Set default route…” checkbox.

- LEFT CLICK the “Restart PPP when connection fails” checkbox.

- Enter any other information pertinent to your connection.



□ Activate interface at boot time

■ Set default route when making connection (defaultroute)

■ Restart PPP when connection fails

Timeout values in seconds for...

no connection:

broken connection:

Maximum packet sizes:

MRU (296-1500):

MTU (296-1500):

Infrequently-used options:

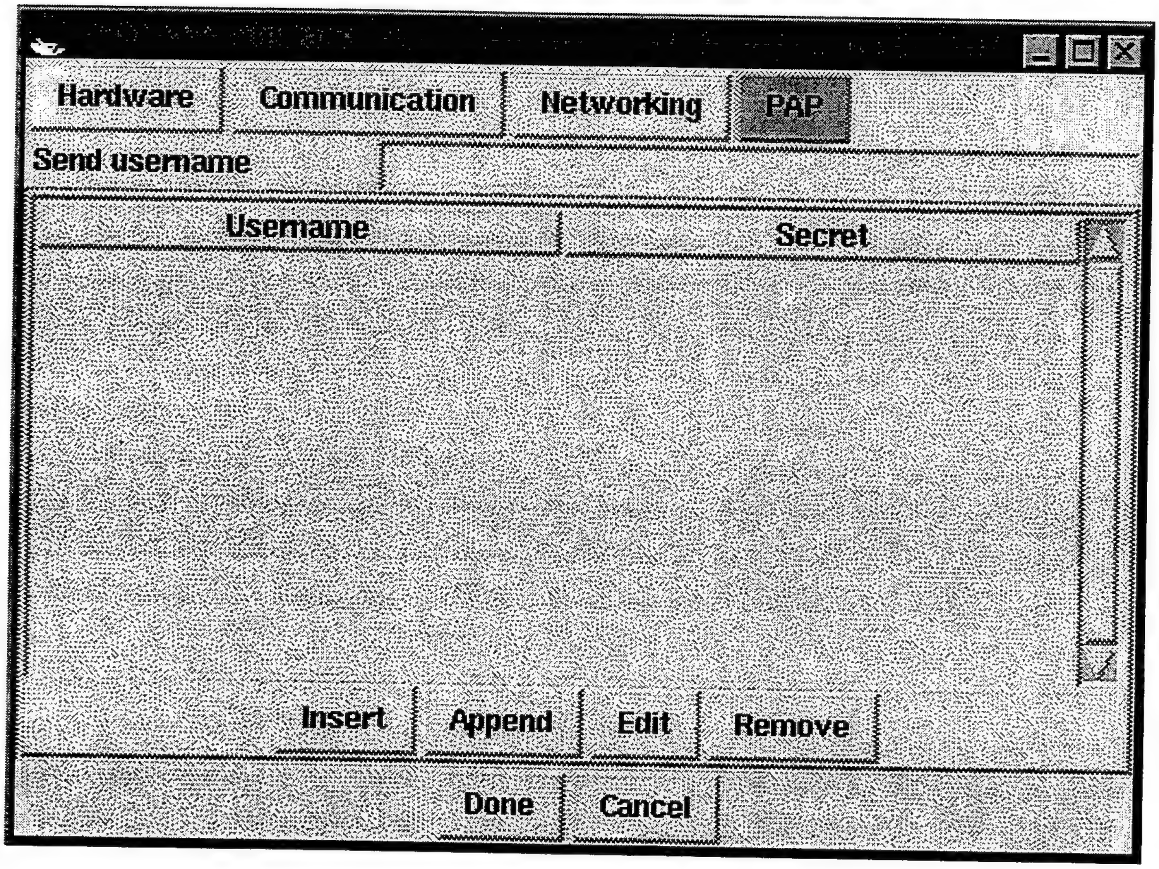
Local IP address:

Remote IP address:

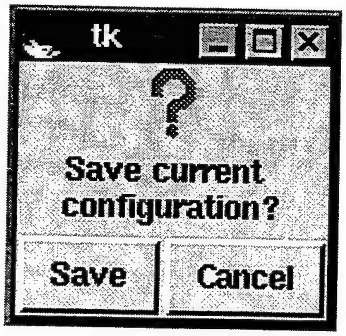
- LEFT CLICK the “PAP” button to open the “Edit PPP Interface-PAP” dialog box.

- The “Edit PPP Interface-PAP” dialog box allows you to enter information specific to your connection to use PAP.

- LEFT CLICK the “Done” button when finished.



- LEFT CLICK the “Save” button to complete the PPP configuration and close the “Edit PPP Interface” dialog box.

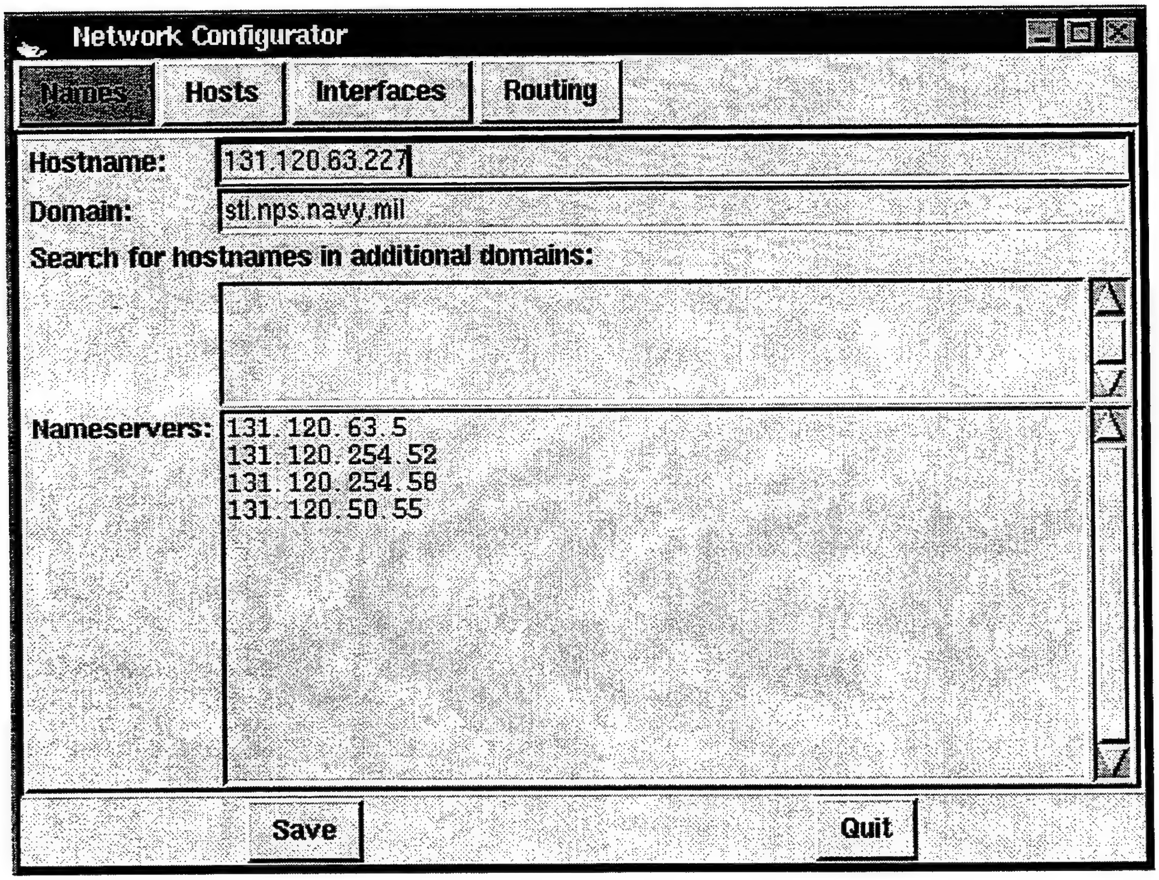


Save current configuration?

2. Configure Red Hat Linux 5.0 to access the Internet using a point-to-point interface.

- LEFT CLICK the "Network" button to open the "Network Configurator-Names" dialog box.

- Enter information specific to your network for "Hostname:", "Domain:", and "Nameservers:".



Hostname: 131.120.63.227

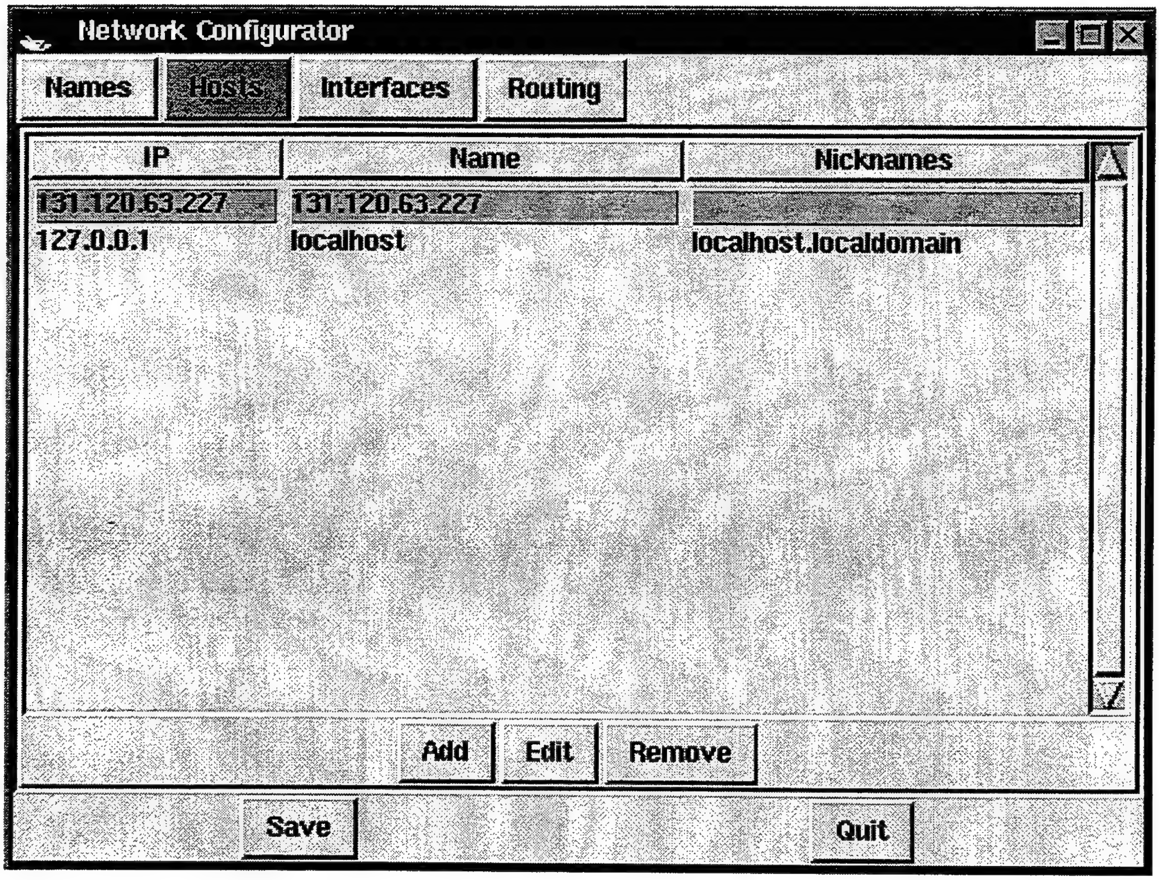
Domain: stl.nps.navy.mil

Search for hostnames in additional domains:

Nameservers:
131.120.63.5
131.120.254.52
131.120.254.58
131.120.50.55

- LEFT CLICK the "Hosts" button to open the "Network Configurator-Hosts" dialog box.

- VERIFY that your IP and hostname are listed. The 127.0.0.1 is a required default that cannot be altered.

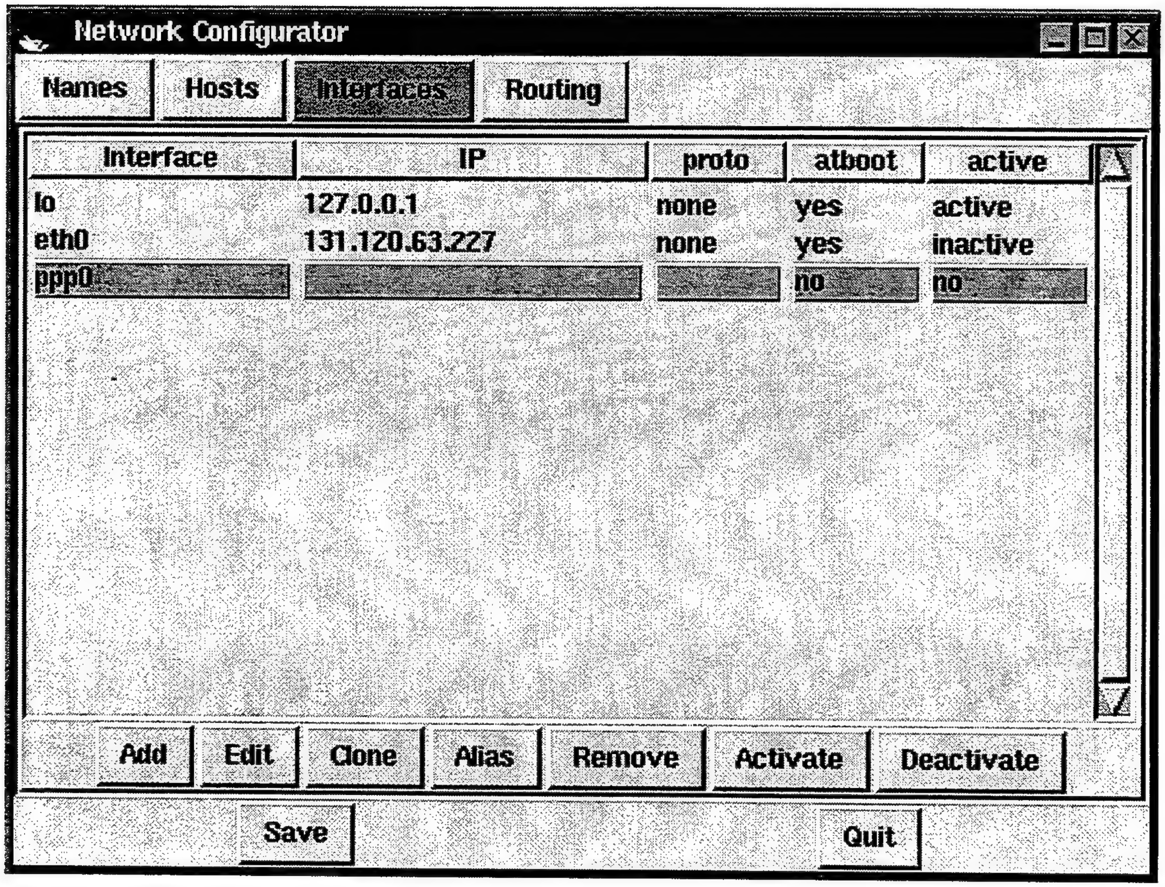


IP	Name	Nicknames
131.120.63.227	131.120.63.227	
127.0.0.1	localhost	localhost.localdomain

- LEFT CLICK the "Interfaces" button to open the "Network Configurator-Interfaces" dialog box.

- VERIFY that the required interfaces are present. This is the same dialog box that was referenced in the previous section for customizing a PPP .

- To VERIFY the settings on a particular interface, LEFT CLICK on that interface and LEFT CLICK the "Edit" button. This will open up the individual interface dialog box so the settings can be changed.



Interface	IP	proto	atboot	active
lo	127.0.0.1	none	yes	active
eth0	131.120.63.227	none	yes	inactive
ppp0			no	no

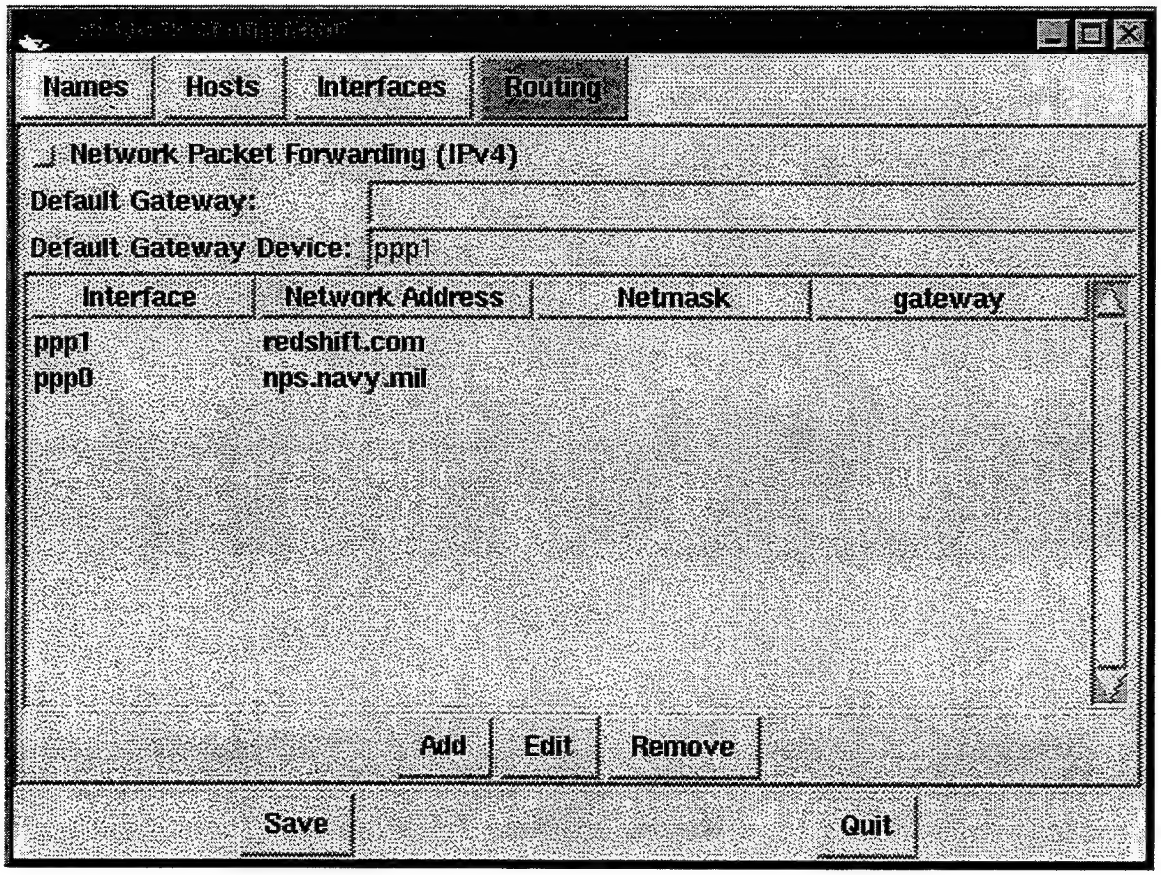
- LEFT CLICK the "Routing" button to open the "Network Configurator-Routing" dialog box.

- Enter the information specific for your network. For a PPP connection the "Default Gateway Device:" must be one of the "pppX" interfaces created earlier. The "ppp0" interface represents the ISP at NPS. "ppp1" represents a commercial ISP in the local area near NPS. For a LAN connection both the "Default Gateway:" and the "Default Gateway Device:" must be included. For example the NPS SeaNet lab would need use the settings listed below

Default Gateway:	131.120.63.1
Default Gateway Device:	eth0 (the NIC for this computer)

- LEFT CLICK "Save" when finished.

- LEFT CLICK "Quit" to close the "Network Configurator" dialog box.



□ Network Packet Forwarding (IPv4)

Default Gateway:

Default Gateway Device: ppp1

Interface	Network Address	Netmask	gateway
ppp1	redshift.com		
ppp0	nps.navy.mil		

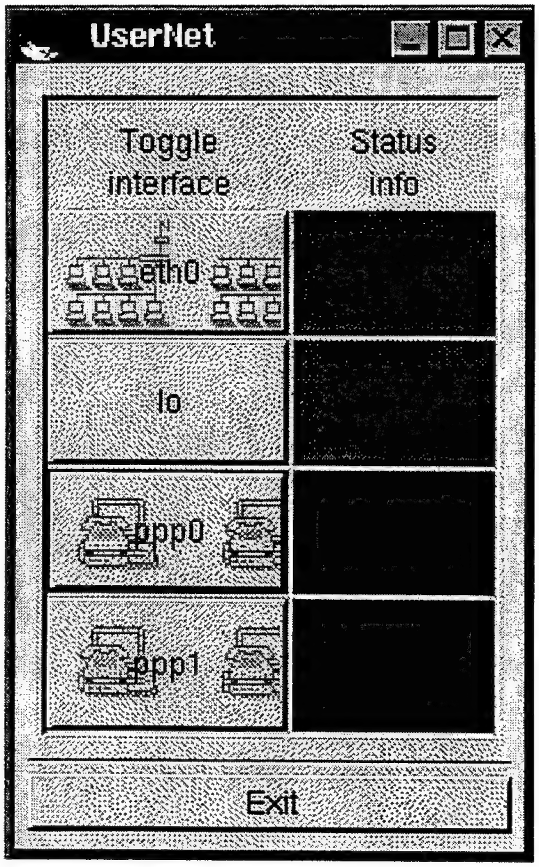
3. Turn specific interfaces on/off using the "Usernet" program.

- LEFT CLICK on any open spot on the desktop or on the "Start" button. This will open the "Start" menu.

- HIGHLIGHT "Programs" to expand the "Programs" submemu.

- HIGHLIGHT "Networking" to expand the "Neworking" submenu

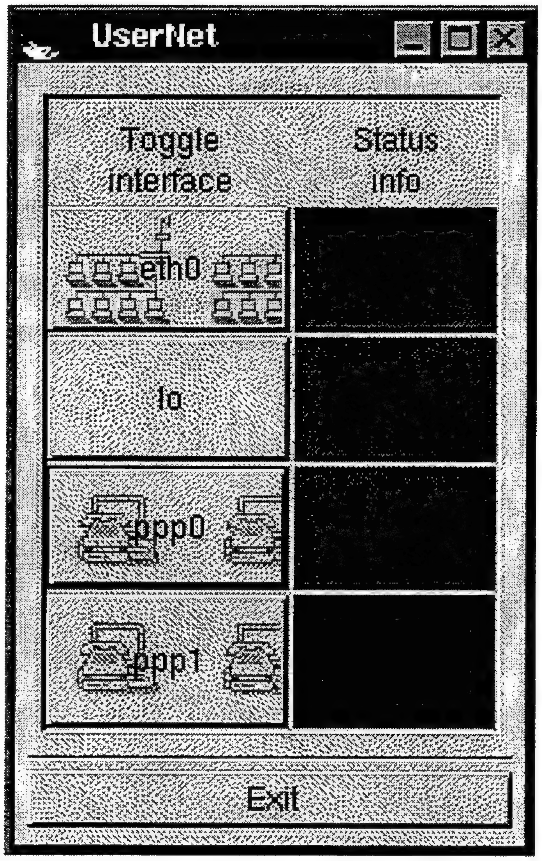
- LEFT CLICK "Usernet" to open the "Usernet" program shown below.



The Usernet program is a simple utility that allows a user to toggle a particular interface on/off with a button named after the interface. The "Status info" provides a color-coded indication whether an interface is active, inactive, or in transition. Here we see that the LAN connection "eth0" has a GREEN status indication and is thus active. The "ppp0" and "ppp1" have RED status indications to show they inactive. The third color is YELLOW to indicate that an interface is in transition. After LEFT CLICKING on a "pppX" interface button there will be a YELLOW status indication as it dials and connects. The status of the interface will turn GREEN when a successful connection with the ISP has been established.

- LEFT CLICK on the desired "pppX" "Toggle Interface" button to initiate the Point-to Point connection. As discussed earlier the status will remain YELLOW until successfully connected when it will turn GREEN as displayed below.

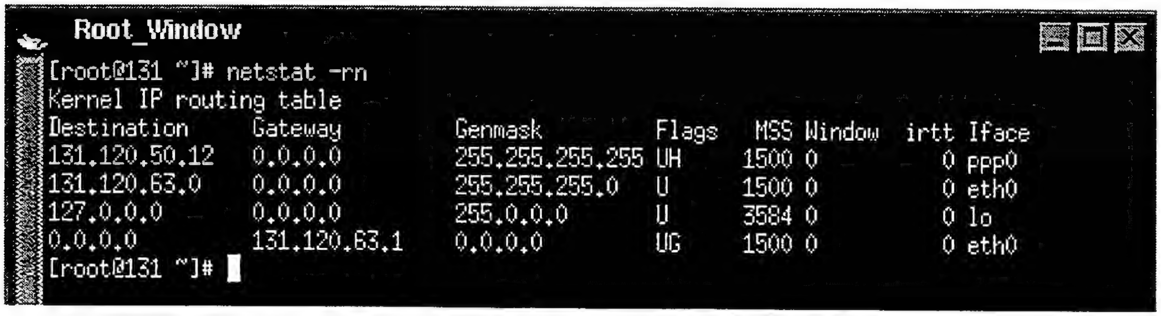
- LEFT CLICK on the desired "pppX" "Toggle Interface" button to terminate the connection.



4. Retrieve a dynamically assigned IP from an ISP using the NESTAT command

- To retrieve a IP address that was dynamically assigned from your ISP, use the NETSTAT command

- TYPE "netstat -rn" from a "Root_Window". From the example below, we can see that for the "ppp0" interface the IP address is "131.120.50 .12".



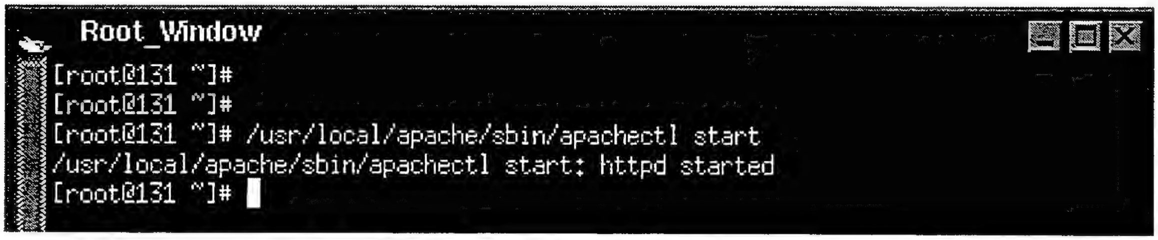
```
[root@131 ~]# netstat -rn
Kernel IP routing table
Destination     Gateway         Genmask         Flags   MSS Window  irtt Iface
131.120.50.12   0.0.0.0         255.255.255.255 UH     1500 0          0 ppp0
131.120.63.0    0.0.0.0         255.255.255.0   U      1500 0          0 eth0
127.0.0.0       0.0.0.0         255.0.0.0       U      3584 0          0 lo
0.0.0.0         131.120.63.1    0.0.0.0         UG     1500 0          0 eth0
[root@131 ~]# █
```

-

5. Launching the Apache HTTP Server

The Apache HTTP server must be configured on the system before it can be used. That configuration is beyond the scope of this document but can be found at http://www.apache.org. Once configured it can be launched using the following command from a "Root_Window." This means using a terminal as the root user. Root indicates superuser privileges and should be used with caution.

- LEFT CLICK on any open spot on the desktop or on the "Start" button. This will open the "Start" menu.
- HIGHLIGHT "System Utilities" to expand the "System Utilities" submenu.
- LEFT CLICK "Root Shell" to open the "Root_Window" terminal.
- TYPE in "/usr/local/apache/sbin/apachectl start" exactly as listed.
- PRESS ENTER. There will be a response "/usr/local/apache/sbin/apachectl start: httpd started" as displayed indicating that the command was successful..



```
[root@131 ~]#
[root@131 ~]#
[root@131 ~]# /usr/local/apache/sbin/apachectl start
/usr/local/apache/sbin/apachectl start: httpd started
[root@131 ~]# █
```

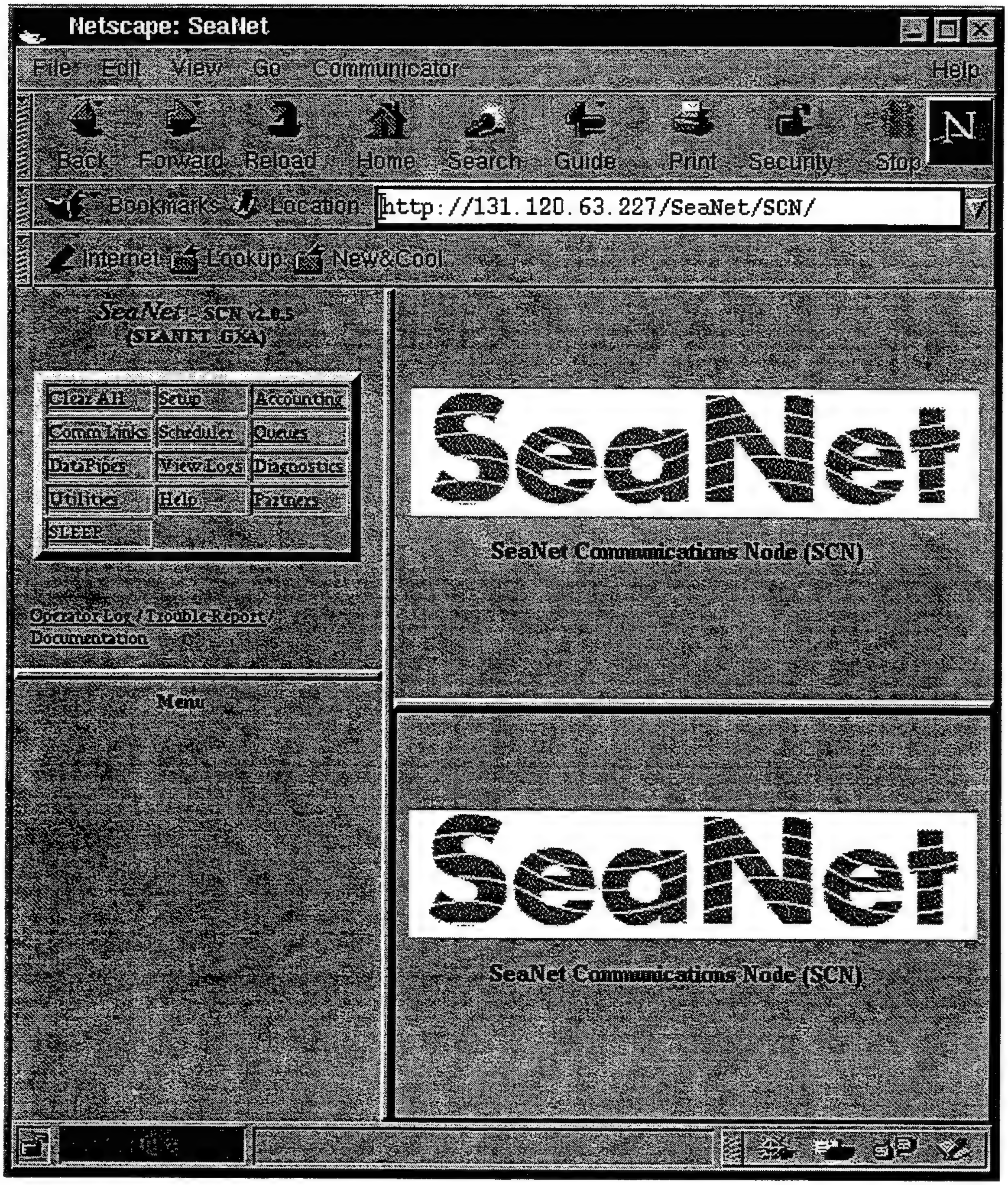
6. Running the SeaNet Communication Node Software

This section will discuss using the SeaNet Communications Node Software and give a brief overview of its functionality. As mentioned earlier in the thesis the SeaNet Communications Node Software is a web browser based program that performs a variety of administrative tasks.

- LEFT CLICK on any open spot on the desktop or on the "Start" button. This will open the "Start" menu.
- HIGHLIGHT "Applications" to expand the "Applications" sub-menu.
- LEFT CLICK "Netscape Navigator" to open the Netscape Navigator program.
-

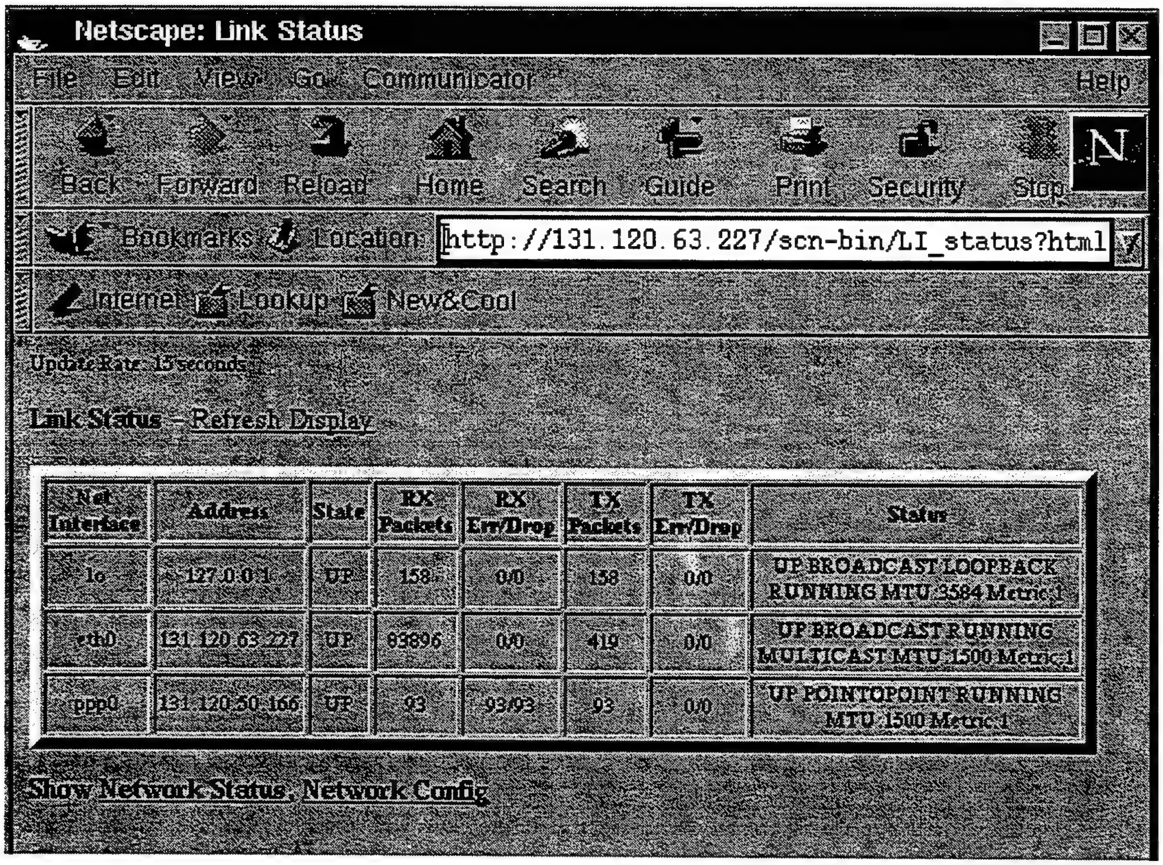
- TYPE “Http://131.120.63.227/SeaNet/SCN” into the address window. NOTE Substitute your own IP in the line above.

- PRESS ENTER when finished. This will open the SeaNet Communications Node Software main screen as shown.



Location: http://131.120.63.227/SeaNet/SCN/

- LEFT CLICK the “Comm Links” hyperlink from the menu box in the top left frame. This will open the “Link Status” frame that is pictured below. This table provide valuable information on the status of communication links.


Netscape: Link Status
File Edit View Go Communicator Help
Back Forward Reload Home Search Guide Print Security Stop
Bookmarks Location:
Internet Lookup New&Cool

http://131.120.63.227/scn-bin/LI_status?html

Update Rate: 15 seconds

Link Status – Refresh Display

Net Interface	Address	State	RX Packets	RX Err/Drop	TX Packets	TX Err/Drop	Status
lo	127.0.0.1	UP	158	0/0	158	0/0	UP BROADCAST LOOPBACK RUNNING MTU 3584 Metric 1
eth0	131.120.63.227	UP	83896	0/0	419	0/0	UP BROADCAST RUNNING MULTICAST MTU 1500 Metric 1
ppp0	131.120.50.166	UP	93	93/93	93	0/0	UP POINTOPOINT RUNNING MTU 1500 Metric 1

Show Network Status, Network Config

- LEFT CLICK the “Accounting” hyperlink to open the “Accounting” frame.

APPENDIX C. LINUX SNMP CONFIGURATION FILE (SNMPD.CONF)

This appendix is a copy of the "snmpd.conf" file. This is the configuration file for the SNMP agent that comes with Red Hat Linux 5.0. This file is located in the "/etc/" directory.

```
#
# snmpd.conf - created Thu Jul 24 22:12:50 MET DST 1997
#

#
# view configuration
#
#       viewName      OID                    included/excluded
#

# Red Hat Special Define for Limited Access.
view    rest          .1.3.6.1.2.1.1.4       included

# internet
view    all           .1.3.6.1               included

# internet
view    mini          .1.3.6.1               included

# for v1 public exclude exclude mib-2.ident.identInfo
# and mib-2.host.hrSWRun in the mini view:
view    xmini         .1.3.6.1               included
view    xmini         .1.3.6.1.2.1.24.1      excluded
view    xmini         .1.3.6.1.2.1.25.4      excluded

# system, snmp, usecAgent, usecStats
view    semi          .1.3.6.1.2.1.1         included
view    semi          .1.3.6.1.2.1.11               included
view    semi          .1.3.6.1.6.3.6.1.1     included
view    semi          .1.3.6.1.6.3.6.1.2     included

# snmp, usecAgent, usecStats
view    semi          .1.3.6.1.2.1.11               included
view    semi          .1.3.6.1.6.3.6.1.1     included
view    semi          .1.3.6.1.6.3.6.1.2     included


#
#
# user configuration
#
#      noneRV  noneWV  authRV  authWV  userName[/authKey]
#
#  If you uncomment the following line:
#      - make sure you understand what giving write access means
#      - use authkey(1) to generate a new hex authkey
#
#user  mini    -       all     all     public/PUT_AUTH_KEY_HERE


#
#
# community configuration
#
#        commName     readV   writeV
#
community public      rest    -
## uncomment for private entry:
```

```
# community private    mini   mini

##
## now follows the specific section of the linux-port.
##

##
## port to use (default is 161):
##
# port:        161

##
## the entry of system.Contact and system.Location:
##
sysContact:            Please configure your snmpd before running.
sysLocation:           Not Configured
## the system name is per default determined from the hostname:
# sysName:             chappell

##
## trap sink address and community string. (passed to snmptrap(1)
## utility). authentraps contains the value of snmpEnableAuthenTraps;
## (default is `disabled').
##
trap sink:             localhost
trap community:               public
snmpEnableAuthenTraps: disabled

##
## specify type and speed of interfaces:
## if the last char is an asterisk, any suffix will match.
## (feel free to add more)
##
interface:     lo0     24     20000000
interface:     eth*     6     10000000
interface:     sl*     28     28800
interface:     ppp*    23     28800
interface:     isdn*   20     64000
interface:     ippp*   20     64000

## end of /etc/snmpd.conf
```

APPENDIX D. LINUX-SUPPORTED MIB VARIABLES

The following list contains all the MIB-II variables supported by Red Hat Linux 5.0. This list was generated by SNMP polling of the NPS shipboard computer with the HPOV NMS. MIB-II object descriptions corresponding to the numbers can be found in RFC 1156 and RFC 1213.

```
Name or IP Address: 131.120.63.227

Community Name:

MIB Object ID: .iso.org.dod.internet.mgmt

MIB Instance:

SNMP Set Value:

MIB Values:
    1.1.1.0 : Linux version 2.0.32 (root@porky.redhat.com) (gcc version 2.7.2.3) #1 Wed
Nov 19 00:46:45 EST 1997
    1.1.2.0 : .iso.org.dod.internet.private.enterprises.1575.1.5
    1.1.3.0 : (10238) 0:01:42.38
    1.1.4.0 : Christopher L. Pratt
    1.1.5.0 : 131.120.63.227
    1.1.6.0 : Root Hall Room 222
    1.1.7.0 : 72
    1.1.8.0 : (10244) 0:01:42.44
    1.1.9.1.2.1 : .iso.org.dod.internet.private.enterprises.1575.1.5.1.1
    1.1.9.1.3.1 : LINUX agent
    1.1.9.1.4.1 : (10248) 0:01:42.48
    1.2.1.0 : 2
    1.2.2.1.1.1 : 1
    1.2.2.1.1.2 : 2
    1.2.2.1.2.1 : lo0
    1.2.2.1.2.2 : eth0
    1.2.2.1.3.1 : 24
    1.2.2.1.3.2 : 6
    1.2.2.1.4.1 : 3584
    1.2.2.1.4.2 : 1500
    1.2.2.1.5.1 : 20000000
    1.2.2.1.5.2 : 10000000
    1.2.2.1.6.1 : 00 00 00 00 00 00
    1.2.2.1.6.2 : 00 C0 4F 86 5D D7
    1.2.2.1.7.1 : 1
    1.2.2.1.7.2 : 1
    1.2.2.1.8.1 : 1
    1.2.2.1.8.2 : 1
    1.2.2.1.9.1 : (0) 0:00:00.00
    1.2.2.1.9.2 : (0) 0:00:00.00
    1.2.2.1.10.1 : 616
    1.2.2.1.10.2 : 281204
    1.2.2.1.11.1 : 2
    1.2.2.1.11.2 : 915
    1.2.2.1.12.1 : 0
    1.2.2.1.12.2 : 0
    1.2.2.1.13.1 : 0
    1.2.2.1.13.2 : 0
    1.2.2.1.14.1 : 0
    1.2.2.1.14.2 : 0
    1.2.2.1.15.1 : 0
    1.2.2.1.15.2 : 0
```

```
1.2.2.1.16.1 : 616
1.2.2.1.16.2 : 189728
1.2.2.1.17.1 : 2
1.2.2.1.17.2 : 618
1.2.2.1.18.1 : 0
1.2.2.1.18.2 : 0
1.2.2.1.19.1 : 0
1.2.2.1.19.2 : 0
1.2.2.1.20.1 : 0
1.2.2.1.20.2 : 0
1.2.2.1.21.1 : 0
1.2.2.1.21.2 : 0
1.2.2.1.22.1 : .iso.org.dod.internet.mgmt.1.2.2.1.22.0.0
1.2.2.1.22.2 : .iso.org.dod.internet.mgmt.1.2.2.1.22.0.0
1.3.1.1.1.2.1.131.120.63.226 : 2
1.3.1.1.2.2.1.131.120.63.226 : 08 00 09 48 7C FF
1.3.1.1.3.2.1.131.120.63.226 : 131.120.63.226
1.4.1.0 : 2
1.4.2.0 : 64
1.4.3.0 : 718
1.4.4.0 : 0
1.4.5.0 : 0
1.4.6.0 : 0
1.4.7.0 : 0
1.4.8.0 : 0
1.4.9.0 : 684
1.4.10.0 : 642
1.4.11.0 : 0
1.4.12.0 : 0
1.4.13.0 : 0
1.4.14.0 : 0
1.4.15.0 : 0
1.4.16.0 : 0
1.4.17.0 : 0
1.4.18.0 : 0
1.4.19.0 : 0
1.4.20.1.1.127.0.0.1 : 127.0.0.1
1.4.20.1.1.131.120.63.227 : 131.120.63.227
1.4.20.1.2.127.0.0.1 : 1
1.4.20.1.2.131.120.63.227 : 2
1.4.20.1.3.127.0.0.1 : 255.0.0.0
1.4.20.1.3.131.120.63.227 : 255.255.255.0
1.4.20.1.4.127.0.0.1 : 1
1.4.20.1.4.131.120.63.227 : 1
1.4.20.1.5.127.0.0.1 : 20480
1.4.20.1.5.131.120.63.227 : 20480
1.4.21.1.1.0.0.0.0 : 0.0.0.0
1.4.21.1.1.127.0.0.0 : 127.0.0.0
1.4.21.1.1.131.120.63.0 : 131.120.63.0
1.4.21.1.2.0.0.0.0 : 2
1.4.21.1.2.127.0.0.0 : 1
1.4.21.1.2.131.120.63.0 : 2
1.4.21.1.3.0.0.0.0 : 1
1.4.21.1.3.127.0.0.0 : 0
1.4.21.1.3.131.120.63.0 : 0
1.4.21.1.4.0.0.0.0 : -1
1.4.21.1.4.127.0.0.0 : -1
1.4.21.1.4.131.120.63.0 : -1
1.4.21.1.5.0.0.0.0 : -1
1.4.21.1.5.127.0.0.0 : -1
1.4.21.1.5.131.120.63.0 : -1
1.4.21.1.6.0.0.0.0 : -1
1.4.21.1.6.127.0.0.0 : -1
1.4.21.1.6.131.120.63.0 : -1
1.4.21.1.7.0.0.0.0 : 131.120.63.1
1.4.21.1.7.127.0.0.0 : 0.0.0.0
1.4.21.1.7.131.120.63.0 : 0.0.0.0
1.4.21.1.8.0.0.0.0 : 4
1.4.21.1.8.127.0.0.0 : 3
1.4.21.1.8.131.120.63.0 : 3
1.4.21.1.9.0.0.0.0 : 2
```

```
1.4.21.1.9.127.0.0.0 : 2
1.4.21.1.9.131.120.63.0 : 2
1.4.21.1.10.0.0.0.0 : 0
1.4.21.1.10.127.0.0.0 : 0
1.4.21.1.10.131.120.63.0 : 0
1.4.21.1.11.0.0.0.0 : 0.0.0.0
1.4.21.1.11.127.0.0.0 : 255.0.0.0
1.4.21.1.11.131.120.63.0 : 255.255.255.0
1.4.21.1.12.0.0.0.0 : -1
1.4.21.1.12.127.0.0.0 : -1
1.4.21.1.12.131.120.63.0 : -1
1.4.22.1.1.2.131.120.63.226 : 2
1.4.22.1.2.2.131.120.63.226 : 08 00 09 48 7C FF
1.4.22.1.3.2.131.120.63.226 : 131.120.63.226
1.4.22.1.4.2.131.120.63.226 : 3
1.5.1.0 : 6
1.5.2.0 : 0
1.5.3.0 : 1
1.5.4.0 : 0
1.5.5.0 : 0
1.5.6.0 : 0
1.5.7.0 : 0
1.5.8.0 : 5
1.5.9.0 : 0
1.5.10.0 : 0
1.5.11.0 : 0
1.5.12.0 : 0
1.5.13.0 : 0
1.5.14.0 : 6
1.5.15.0 : 0
1.5.16.0 : 1
1.5.17.0 : 0
1.5.18.0 : 0
1.5.19.0 : 0
1.5.20.0 : 0
1.5.21.0 : 0
1.5.22.0 : 5
1.5.23.0 : 0
1.5.24.0 : 0
1.5.25.0 : 0
1.5.26.0 : 0
1.6.1.0 : 1
1.6.2.0 : 0
1.6.3.0 : 0
1.6.4.0 : 0
1.6.5.0 : 0
1.6.6.0 : 0
1.6.7.0 : 0
1.6.8.0 : 0
1.6.9.0 : 0
1.6.10.0 : 0
1.6.11.0 : 0
1.6.12.0 : 0
1.6.13.1.1.0.0.0.0.21.0.0.0.0.0 : 2
1.6.13.1.1.0.0.0.0.22.0.0.0.0.0 : 2
1.6.13.1.1.0.0.0.0.23.0.0.0.0.0 : 2
1.6.13.1.1.0.0.0.0.111.0.0.0.0.0 : 2
1.6.13.1.1.0.0.0.0.139.0.0.0.0.0 : 2
1.6.13.1.1.0.0.0.0.515.0.0.0.0.0 : 2
1.6.13.1.1.0.0.0.0.5432.0.0.0.0.0 : 2
1.6.13.1.2.0.0.0.0.21.0.0.0.0.0 : 0.0.0.0
1.6.13.1.2.0.0.0.0.22.0.0.0.0.0 : 0.0.0.0
1.6.13.1.2.0.0.0.0.23.0.0.0.0.0 : 0.0.0.0
1.6.13.1.2.0.0.0.0.111.0.0.0.0.0 : 0.0.0.0
1.6.13.1.2.0.0.0.0.139.0.0.0.0.0 : 0.0.0.0
1.6.13.1.2.0.0.0.0.515.0.0.0.0.0 : 0.0.0.0
1.6.13.1.2.0.0.0.0.5432.0.0.0.0.0 : 0.0.0.0
1.6.13.1.3.0.0.0.0.21.0.0.0.0.0 : 21
1.6.13.1.3.0.0.0.0.22.0.0.0.0.0 : 22
1.6.13.1.3.0.0.0.0.23.0.0.0.0.0 : 23
1.6.13.1.3.0.0.0.0.111.0.0.0.0.0 : 111
```

```
1.6.13.1.3.0.0.0.0.139.0.0.0.0.0 : 139
1.6.13.1.3.0.0.0.0.515.0.0.0.0.0 : 515
1.6.13.1.3.0.0.0.0.5432.0.0.0.0.0 : 5432
1.6.13.1.4.0.0.0.0.21.0.0.0.0.0 : 0.0.0.0
1.6.13.1.4.0.0.0.0.22.0.0.0.0.0 : 0.0.0.0
1.6.13.1.4.0.0.0.0.23.0.0.0.0.0 : 0.0.0.0
1.6.13.1.4.0.0.0.0.111.0.0.0.0.0 : 0.0.0.0
1.6.13.1.4.0.0.0.0.139.0.0.0.0.0 : 0.0.0.0
1.6.13.1.4.0.0.0.0.515.0.0.0.0.0 : 0.0.0.0
1.6.13.1.4.0.0.0.0.5432.0.0.0.0.0 : 0.0.0.0
1.6.13.1.5.0.0.0.0.21.0.0.0.0.0 : 0
1.6.13.1.5.0.0.0.0.22.0.0.0.0.0 : 0
1.6.13.1.5.0.0.0.0.23.0.0.0.0.0 : 0
1.6.13.1.5.0.0.0.0.111.0.0.0.0.0 : 0
1.6.13.1.5.0.0.0.0.139.0.0.0.0.0 : 0
1.6.13.1.5.0.0.0.0.515.0.0.0.0.0 : 0
1.6.13.1.5.0.0.0.0.5432.0.0.0.0.0 : 0
1.7.1.0 : 769
1.7.2.0 : 1
1.7.3.0 : 0
1.7.4.0 : 772
1.7.5.1.1.0.0.0.0.111 : 0.0.0.0
1.7.5.1.1.0.0.0.0.161 : 0.0.0.0
1.7.5.1.1.0.0.0.0.514 : 0.0.0.0
1.7.5.1.2.0.0.0.0.111 : 111
1.7.5.1.2.0.0.0.0.161 : 161
1.7.5.1.2.0.0.0.0.514 : 514
1.11.1.0 : 779
1.11.2.0 : 779
1.11.3.0 : 0
1.11.4.0 : 0
1.11.5.0 : 0
1.11.6.0 : 0
1.11.8.0 : 0
1.11.9.0 : 0
1.11.10.0 : 0
1.11.11.0 : 0
1.11.12.0 : 0
1.11.13.0 : 790
1.11.14.0 : 0
1.11.15.0 : 8
1.11.16.0 : 785
1.11.17.0 : 0
1.11.18.0 : 0
1.11.19.0 : 0
1.11.20.0 : 0
1.11.21.0 : 0
1.11.22.0 : 0
1.11.24.0 : 0
1.11.25.0 : 0
1.11.26.0 : 0
1.11.27.0 : 0
1.11.28.0 : 0
1.11.29.0 : 0
1.11.30.0 : 1
1.24.1.1.1.1.0.0.0.0.21.0.0.0.0.0 : 1
1.24.1.1.1.1.0.0.0.0.22.0.0.0.0.0 : 1
1.24.1.1.1.1.0.0.0.0.23.0.0.0.0.0 : 1
1.24.1.1.1.1.0.0.0.0.111.0.0.0.0.0 : 1
1.24.1.1.1.1.0.0.0.0.139.0.0.0.0.0 : 1
1.24.1.1.1.1.0.0.0.0.515.0.0.0.0.0 : 1
1.24.1.1.1.1.0.0.0.0.5432.0.0.0.0.0 : 1
1.24.1.1.1.2.0.0.0.0.21.0.0.0.0.0 : unix
1.24.1.1.1.2.0.0.0.0.22.0.0.0.0.0 : unix
1.24.1.1.1.2.0.0.0.0.23.0.0.0.0.0 : unix
1.24.1.1.1.2.0.0.0.0.111.0.0.0.0.0 : unix
1.24.1.1.1.2.0.0.0.0.139.0.0.0.0.0 : unix
1.24.1.1.1.2.0.0.0.0.515.0.0.0.0.0 : unix
1.24.1.1.1.2.0.0.0.0.5432.0.0.0.0.0 : unix
1.24.1.1.1.3.0.0.0.0.21.0.0.0.0.0 : US-ASCII
1.24.1.1.1.3.0.0.0.0.22.0.0.0.0.0 : US-ASCII
```

```
1.24.1.1.1.3.0.0.0.0.23.0.0.0.0.0 : US-ASCII
1.24.1.1.1.3.0.0.0.0.111.0.0.0.0.0 : US-ASCII
1.24.1.1.1.3.0.0.0.0.139.0.0.0.0.0 : US-ASCII
1.24.1.1.1.3.0.0.0.0.515.0.0.0.0.0 : US-ASCII
1.24.1.1.1.3.0.0.0.0.5432.0.0.0.0.0 : US-ASCII
1.24.1.1.1.4.0.0.0.0.21.0.0.0.0.0 : root
1.24.1.1.1.4.0.0.0.0.22.0.0.0.0.0 : root
1.24.1.1.1.4.0.0.0.0.23.0.0.0.0.0 : root
1.24.1.1.1.4.0.0.0.0.111.0.0.0.0.0 : root
1.24.1.1.1.4.0.0.0.0.139.0.0.0.0.0 : root
1.24.1.1.1.4.0.0.0.0.515.0.0.0.0.0 : root
1.24.1.1.1.4.0.0.0.0.5432.0.0.0.0.0 : postgres
1.24.1.1.1.5.0.0.0.0.21.0.0.0.0.0 : root,,,,
1.24.1.1.1.5.0.0.0.0.22.0.0.0.0.0 : root,,,,
1.24.1.1.1.5.0.0.0.0.23.0.0.0.0.0 : root,,,,
1.24.1.1.1.5.0.0.0.0.111.0.0.0.0.0 : root,,,,
1.24.1.1.1.5.0.0.0.0.139.0.0.0.0.0 : root,,,,
1.24.1.1.1.5.0.0.0.0.515.0.0.0.0.0 : root,,,,
1.24.1.1.1.5.0.0.0.0.5432.0.0.0.0.0 : PostreSQL Server
1.25.1.1.0 : (12446) 0:02:04.46
1.25.1.2.0 : 00 62 08 1B 0D 2B 1E 00
1.25.1.3.0 : 774
1.25.1.4.0 : auto BOOT_IMAGE=linux ro root=306
1.25.1.5.0 : 2
1.25.1.6.0 : 24
1.25.1.7.0 : 512
1.25.2.2.0 : 65536
1.25.2.3.1.1.1 : 1
1.25.2.3.1.1.2 : 2
1.25.2.3.1.1.774 : 774
1.25.2.3.1.2.1 : .iso.org.dod.internet.mgmt.1.25.2.1.2
1.25.2.3.1.2.2 : .iso.org.dod.internet.mgmt.1.25.2.1.3
1.25.2.3.1.2.774 : .iso.org.dod.internet.mgmt.1.25.2.1.4
1.25.2.3.1.3.1 : Mem
1.25.2.3.1.3.2 : Swap
1.25.2.3.1.3.774 : Disk
1.25.2.3.1.4.1 : 1024
1.25.2.3.1.4.2 : 1024
1.25.2.3.1.4.774 : 1024
1.25.2.3.1.5.1 : 63160
1.25.2.3.1.5.2 : 64224
1.25.2.3.1.5.774 : 987220
1.25.2.3.1.6.1 : 11780
1.25.2.3.1.6.2 : 0
1.25.2.3.1.6.774 : 760413
1.25.2.3.1.7.1 : 0
1.25.2.3.1.7.2 : 0
1.25.2.3.1.7.774 : 0
1.25.3.2.1.1.1 : 1
1.25.3.2.1.2.1 : .iso.org.dod.internet.mgmt.1.25.3.1.3
1.25.3.2.1.3.1 : processor: 0, cpu: 686
1.25.3.2.1.4.1 : .ccitt.0
1.25.3.2.1.5.1 : 2
1.25.3.2.1.6.1 : 0
1.25.3.3.1.1.1 : .ccitt.0
1.25.3.3.1.2.1 : 0
1.25.3.8.1.1.774 : 774
1.25.3.8.1.2.774 : /dev/hda6
1.25.3.8.1.3.774 : /
1.25.3.8.1.4.774 : .iso.org.dod.internet.mgmt.1.25.3.9.2
1.25.3.8.1.5.774 : 1
1.25.3.8.1.6.774 : 1
1.25.3.8.1.7.774 : 774
1.25.3.8.1.8.774 : 00 00 01 01 00 00 00 00
1.25.3.8.1.9.774 : 00 00 01 01 00 00 00 00
1.25.4.1.0 : 1
1.25.4.2.1.1.0 : 0
1.25.4.2.1.1.1 : 1
1.25.4.2.1.1.2 : 2
1.25.4.2.1.1.3 : 3
1.25.4.2.1.1.23 : 23
```

```
1.25.4.2.1.1.180 : 180
1.25.4.2.1.1.189 : 189
1.25.4.2.1.1.200 : 200
1.25.4.2.1.1.211 : 211
1.25.4.2.1.1.222 : 222
1.25.4.2.1.1.233 : 233
1.25.4.2.1.1.246 : 246
1.25.4.2.1.1.257 : 257
1.25.4.2.1.1.268 : 268
1.25.4.2.1.1.283 : 283
1.25.4.2.1.1.288 : 288
1.25.4.2.1.1.304 : 304
1.25.4.2.1.1.326 : 326
1.25.4.2.1.1.327 : 327
1.25.4.2.1.1.328 : 328
1.25.4.2.1.1.329 : 329
1.25.4.2.1.1.330 : 330
1.25.4.2.1.1.331 : 331
1.25.4.2.1.1.333 : 333
1.25.4.2.1.1.334 : 334
1.25.4.2.1.2.0 : (unknown)
1.25.4.2.1.2.1 : (unknown)
1.25.4.2.1.2.2 : (unknown)
1.25.4.2.1.2.3 : (unknown)
1.25.4.2.1.2.23 : (unknown)
1.25.4.2.1.2.180 : (unknown)
1.25.4.2.1.2.189 : (unknown)
1.25.4.2.1.2.200 : (unknown)
1.25.4.2.1.2.211 : (unknown)
1.25.4.2.1.2.222 : (unknown)
1.25.4.2.1.2.233 : (unknown)
1.25.4.2.1.2.246 : (unknown)
1.25.4.2.1.2.257 : (unknown)
1.25.4.2.1.2.268 : (unknown)
1.25.4.2.1.2.283 : (unknown)
1.25.4.2.1.2.288 : (unknown)
1.25.4.2.1.2.304 : (unknown)
1.25.4.2.1.2.326 : (unknown)
1.25.4.2.1.2.327 : (unknown)
1.25.4.2.1.2.328 : (unknown)
1.25.4.2.1.2.329 : (unknown)
1.25.4.2.1.2.330 : (unknown)
1.25.4.2.1.2.331 : (unknown)
1.25.4.2.1.2.333 : (unknown)
1.25.4.2.1.2.334 : (unknown)
1.25.4.2.1.3.0 : .ccitt.0
1.25.4.2.1.3.1 : .ccitt.0
1.25.4.2.1.3.2 : .ccitt.0
1.25.4.2.1.3.3 : .ccitt.0
1.25.4.2.1.3.23 : .ccitt.0
1.25.4.2.1.3.180 : .ccitt.0
1.25.4.2.1.3.189 : .ccitt.0
1.25.4.2.1.3.200 : .ccitt.0
1.25.4.2.1.3.211 : .ccitt.0
1.25.4.2.1.3.222 : .ccitt.0
1.25.4.2.1.3.233 : .ccitt.0
1.25.4.2.1.3.246 : .ccitt.0
1.25.4.2.1.3.257 : .ccitt.0
1.25.4.2.1.3.268 : .ccitt.0
1.25.4.2.1.3.283 : .ccitt.0
1.25.4.2.1.3.288 : .ccitt.0
1.25.4.2.1.3.304 : .ccitt.0
1.25.4.2.1.3.326 : .ccitt.0
1.25.4.2.1.3.327 : .ccitt.0
1.25.4.2.1.3.328 : .ccitt.0
1.25.4.2.1.3.329 : .ccitt.0
1.25.4.2.1.3.330 : .ccitt.0
1.25.4.2.1.3.331 : .ccitt.0
1.25.4.2.1.3.333 : .ccitt.0
1.25.4.2.1.3.334 : .ccitt.0
1.25.4.2.1.4.0 : (unknown)
```

```
1.25.4.2.1.4.1 : init
1.25.4.2.1.4.2 : kflushd
1.25.4.2.1.4.3 : kswapd
1.25.4.2.1.4.23 : kerneld
1.25.4.2.1.4.180 : syslogd
1.25.4.2.1.4.189 : klogd
1.25.4.2.1.4.200 : atd
1.25.4.2.1.4.211 : crond
1.25.4.2.1.4.222 : portmap
1.25.4.2.1.4.233 : snmpd
1.25.4.2.1.4.246 : inetd
1.25.4.2.1.4.257 : sshd
1.25.4.2.1.4.268 : lpd
1.25.4.2.1.4.283 : gpm
1.25.4.2.1.4.288 : postmaster
1.25.4.2.1.4.304 : smbd
1.25.4.2.1.4.326 : login
1.25.4.2.1.4.327 : mingetty
1.25.4.2.1.4.328 : mingetty
1.25.4.2.1.4.329 : mingetty
1.25.4.2.1.4.330 : mingetty
1.25.4.2.1.4.331 : mingetty
1.25.4.2.1.4.333 : update
1.25.4.2.1.4.334 : tcsh
1.25.4.2.1.5.0 : (unknown)
1.25.4.2.1.5.1 : init [3]
1.25.4.2.1.5.2 : (kflushd)
1.25.4.2.1.5.3 : (kswapd)
1.25.4.2.1.5.23 : /sbin/kerneld
1.25.4.2.1.5.180 : syslogd
1.25.4.2.1.5.189 : klogd
1.25.4.2.1.5.200 : /usr/sbin/atd
1.25.4.2.1.5.211 : crond
1.25.4.2.1.5.222 : portmap
1.25.4.2.1.5.233 : /usr/sbin/snmpd -f
1.25.4.2.1.5.246 : inetd
1.25.4.2.1.5.257 : /usr/local/sbin/sshd
1.25.4.2.1.5.268 : lpd
1.25.4.2.1.5.283 : gpm -t PS/2
1.25.4.2.1.5.288 : /usr/bin/postmaster -S -D/var/lib/pgsql
1.25.4.2.1.5.304 : smbd -D
1.25.4.2.1.5.326 : /bin/login -- root
1.25.4.2.1.5.327 : /sbin/mingetty tty2
1.25.4.2.1.5.328 : /sbin/mingetty tty3
1.25.4.2.1.5.329 : /sbin/mingetty tty4
1.25.4.2.1.5.330 : /sbin/mingetty tty5
1.25.4.2.1.5.331 : /sbin/mingetty tty6
1.25.4.2.1.5.333 : update (bdflush)
1.25.4.2.1.5.334 : -tcsh
1.25.4.2.1.6.0 : 4
1.25.4.2.1.6.1 : 4
1.25.4.2.1.6.2 : 4
1.25.4.2.1.6.3 : 4
1.25.4.2.1.6.23 : 4
1.25.4.2.1.6.180 : 4
1.25.4.2.1.6.189 : 4
1.25.4.2.1.6.200 : 4
1.25.4.2.1.6.211 : 4
1.25.4.2.1.6.222 : 4
1.25.4.2.1.6.233 : 4
1.25.4.2.1.6.246 : 4
1.25.4.2.1.6.257 : 4
1.25.4.2.1.6.268 : 4
1.25.4.2.1.6.283 : 4
1.25.4.2.1.6.288 : 4
1.25.4.2.1.6.304 : 4
1.25.4.2.1.6.326 : 4
1.25.4.2.1.6.327 : 4
1.25.4.2.1.6.328 : 4
1.25.4.2.1.6.329 : 4
1.25.4.2.1.6.330 : 4
```

```
1.25.4.2.1.6.331 : 4
1.25.4.2.1.6.333 : 4
1.25.4.2.1.6.334 : 4
1.25.4.2.1.7.0 : 2
1.25.4.2.1.7.1 : 2
1.25.4.2.1.7.2 : 2
1.25.4.2.1.7.3 : 2
1.25.4.2.1.7.23 : 2
1.25.4.2.1.7.180 : 2
1.25.4.2.1.7.189 : 2
1.25.4.2.1.7.200 : 2
1.25.4.2.1.7.211 : 2
1.25.4.2.1.7.222 : 2
1.25.4.2.1.7.233 : 1
1.25.4.2.1.7.246 : 2
1.25.4.2.1.7.257 : 2
1.25.4.2.1.7.268 : 2
1.25.4.2.1.7.283 : 2
1.25.4.2.1.7.288 : 2
1.25.4.2.1.7.304 : 2
1.25.4.2.1.7.326 : 2
1.25.4.2.1.7.327 : 2
1.25.4.2.1.7.328 : 2
1.25.4.2.1.7.329 : 2
1.25.4.2.1.7.330 : 2
1.25.4.2.1.7.331 : 2
1.25.4.2.1.7.333 : 2
1.25.4.2.1.7.334 : 2
1.25.5.1.1.1.0 : 0
1.25.5.1.1.1.1 : 337
1.25.5.1.1.1.2 : 0
1.25.5.1.1.1.3 : 0
1.25.5.1.1.1.23 : 1
1.25.5.1.1.1.180 : 7
1.25.5.1.1.1.189 : 11
1.25.5.1.1.1.200 : 1
1.25.5.1.1.1.211 : 1
1.25.5.1.1.1.222 : 1
1.25.5.1.1.1.233 : 36
1.25.5.1.1.1.246 : 1
1.25.5.1.1.1.257 : 107
1.25.5.1.1.1.268 : 0
1.25.5.1.1.1.283 : 0
1.25.5.1.1.1.288 : 0
1.25.5.1.1.1.304 : 0
1.25.5.1.1.1.326 : 10
1.25.5.1.1.1.327 : 1
1.25.5.1.1.1.328 : 0
1.25.5.1.1.1.329 : 0
1.25.5.1.1.1.330 : 1
1.25.5.1.1.1.331 : 1
1.25.5.1.1.1.333 : 1
1.25.5.1.1.1.334 : 11
1.25.5.1.1.2.0 : 0
1.25.5.1.1.2.1 : 400
1.25.5.1.1.2.2 : 0
1.25.5.1.1.2.3 : 0
1.25.5.1.1.2.23 : 356
1.25.5.1.1.2.180 : 452
1.25.5.1.1.2.189 : 548
1.25.5.1.1.2.200 : 420
1.25.5.1.1.2.211 : 492
1.25.5.1.1.2.222 : 340
1.25.5.1.1.2.233 : 764
1.25.5.1.1.2.246 : 416
1.25.5.1.1.2.257 : 572
1.25.5.1.1.2.268 : 428
1.25.5.1.1.2.283 : 352
1.25.5.1.1.2.288 : 816
1.25.5.1.1.2.304 : 600
1.25.5.1.1.2.326 : 940
```

```
1.25.5.1.1.2.327 : 320
1.25.5.1.1.2.328 : 320
1.25.5.1.1.2.329 : 320
1.25.5.1.1.2.330 : 320
1.25.5.1.1.2.331 : 320
1.25.5.1.1.2.333 : 244
1.25.5.1.1.2.334 : 868
```

LIST OF REFERENCES

[1] *SeaNet Project Homepage*, "SeaNet: Extending the Internet to the Oceans," (http://www.seanet.int), August 1998.

[2] Comer, Douglas E., *Computer Networks and Internets*, Prentice-Hall, Inc., 1997.

[3] Dodsworth, Clark, *Digital Illusions*, Addison-Wesley Publishing Company, Inc., 1997.

[4] Stallings, William, *SNMP, SNMPv2 and RMON*, Addison-Wesley Publishing Company, Inc., 1996.

[5] Andalis, Eric L., *Web-Based Network Management Tools For U.S. Navy Mission-Centric Applications*, Masters Thesis, Naval Postgraduate School, September, 1997.

[6] Harrington, David, "SNMP Version 3," *The Simple Times* (http://ww.simple-times.org/pub/simple-times/issues/5-1.html), December, 1997.

[7] *Information Technology Standards Guidance (ITSG)*, Version 98-1.1, Department of the Navy Chief Information Officer ITSG Integrated Product Team, Department of the Navy, 15 June 1998.

[8] "SNMP Version 3", *The Internet Engineering Task Force Home Page* (http://www.ietf.org/html.charters/snmpv3-charter.html), 14 September 1998.

[9] *Request for Comment 1213, Management Information Base for Network Management of TCP/IP-based internets: MIB-II*, K. McCloghrie and M.T. Rose, Network Working Group, (http://nic.mil/ftp/rfc/rfc1213.txt), 1991.

[10] Harnedy, Sean J., *Total SNMP: Exploring the Simple Network Management Protocol*, Prentice-Hall, Inc., 1997.

[11] Huntington-Lee, Jill, Kornel Terplan and Jeffrey A. Gibson, *HP OpenView: A Manager's Guide*, McGraw-Hill Companies, Inc., 1996.

[12] Reed, Kenneth, *Data Network Handbook: An interactive Guide to network Architecture and Operations*, Van Nostrand Reinhold, 1996.

[13] Comer, Douglas, E., Internetworking with TCP/IP, Volume I: Principles, Protocols, and Architecture, Prentice-Hall, Inc., 1995.

[14] Feibel, Werner, *Encyclopedia of Networking, Second Edition*, Sybex Inc., 1996.

[15] KVH Industries, Inc., Tracphone 50 brochure, (http://www.kvh.com), 1998.

[16] American Mobile, (http://www.skycell.com), 1998.

[17] *Hewlett-Packard OpenView Windows User's Guide*, Manual Part Number: J231-90002, Hewlett-Packard Co., December 1993.

[18] *Request for Comment 1156, Management Information Base for Network Management of TCP/IP-based internets*, K. McCloghrie and M.T. Rose, Network Working Group, (http://nic.mil/ftp/rfc/rfc1156.txt), 1990.

[19] Welsh, Matt, Lar Kaufman, *Running Linux*, O'Reilly& Associates, Inc., 1996.

INITIAL DISTRIBUTION LIST

1. Defense Technical Information Center 2
   8725 John J. Kingman Rd., STE 0944
   Ft. Belvoir, VA 22060-6218

2. Dudley Knox Library 2
   Naval Postgraduate School
   411 Dyer Rd.
   Monterey, Ca 93943-5101

3. Professor Rex Buddenberg 1
   Department of Systems Management (Code SM/Bu)
   Naval Postgraduate School
   Monterey, CA 93943

4. Professor Don Brutzman 1
   Undersea Warfare Academic Group (Code UW/Br)
   Naval Postgraduate School
   Monterey, CA 93943

5. Howard Greene 1
   American Mobile Corp.
   10802 Parkridge Blvd.
   Reston, VA 20191

6. Robert Heinmiller 2
   Omnet, Inc.
   PO Box 1285
   Staunton, VA 24402

7. Ellen Kappel 1
   Joint Oceanographic Institutions
   1755 Massachusetts Avenue, NW, Suite 800
   Washington, DC 20036-2102

8. Andrew Maffei/Steve Lerner 2
   Applied Ocean Physics & Engineering Dept.
   Woods Hole Oceanographic Institution
   Woods Hole, MA 02543

9. Dale Chayes/Richard Perry 2
Lamont Doherty Earth Observatory
Route 9W
Palisades, NY 10964

10. Bill Jacobs 1
NCCOSC RDTE Div
53560 Hull Street
San Diego, CA 92152-5001

11. Andy Givens 1
U.S. Coast Guard Information Systems Center
7323 Telegraph Road
Alexandria, VA 22310-3999

12. Vint Cerf.. 1
Corporation for National Research Initiatives
1895 Preston White Drive, Suite 100
Reston, VA 22091

13. Gordon Shillito 1
1015 Chestnut Ridge Drive
Martinsburg, WV 2540

14. LT Chris Pratt.. 1
25 Sky View Road
Cohoes, NY 12047

15. LT Don Murray 1
Route 1 Box 9
Freedom, OK 73842